

**ANÁLISIS DE CARACTERÍSTICAS TIEMPO FRECUENCIA PARA LA
CLASIFICACIÓN DE LENGUAJE DE SEÑAS COLOMBIANO
UTILIZANDO SENSORES INVASIVOS.**

Julio Alberto Diaz Triviño

Tutor

M.Sc. Arley Bejarano Martínez

**Universidad Tecnológica de Pereira
Facultad de ingenierías Eléctrica, electrónica, física y ciencias de la computación
Programa de Ingeniería Electrónica**

CONTENIDO.

1. INTRODUCCIÓN	12
2. PLANTEAMIENTO DEL PROBLEMA	13
3. JUSTIFICACIÓN.....	16
4. OBJETIVO GENERAL.....	17
5. OBJETIVOS ESPECÍFICOS.....	17
6. ESTADO DEL ARTE	18
7. MARCO TEÓRICO.....	20
7.1 Sensores	20
7.1.1 Sensor de electromiografía	21
7.1.2 Unidad de medición inercial (IMU):	21
7.2 Descriptores Tiempo-Frecuencia	22
7.2.1 Transformada rápida de Fourier (FFT)	22
7.2.2 Transformada Wavelet.....	24
7.2.3 Transformada corta de Fourier	25
7.3 Características	27
7.3.1 Media	27
7.3.2 Desviación estándar	27
7.3.3 Curtosis.....	27
7.3.4 Sesgo	28
7.4 Aprendizaje maquina (Machine Learning)	29
7.4.1 Maquina de soporte vectorial (SVM)	29
7.4.2 Redes Neuronales	30
7.5 Análisis de componentes.....	31
7.5.1 Relieff	31
7.5.2 PCA (Principal Component Analysis)	31

7.5.3 Correlación	32
8. DESARROLLO	33
8.1 Base de datos	33
8.1.1 Señas	33
8.1.2 Toma de señas.....	34
8.2 Aplicación de descriptores tiempo-frecuencia	36
8.2.1 FFT	36
8.2.2 Wavelet	38
8.2.3 STFT	38
8.3 Toma de características.....	39
8.4 Aprendizaje maquina	40
8.4.1 SVM	40
8.4.2 Redes neuronales	41
8.5 Análisis de componentes.....	42
8.5.1 PCA (Principal Component Analysis)	42
8.5.2 Correlación	43
8.5.3 Relieff	44
9. ANÁLISIS Y RESULTADOS	45
9.1 Resultados obtenidos para SVM	46
9.1.1 SVM utilizando FFT	46
9.2 Resultados obtenidos para Redes Neuronales	61
9.3 Resultados obtenidos sobre el Análisis de componentes	62
9.3.1 Resultados para FFT	62
9.3.2 Resultados para Wavelet	64
9.3.3 Resultados para STFT	65

9.4 Comparación de resultados de Aprendizaje Maquina antes y después del análisis de componentes	67
9.4.1 Antes y después para FFT aplicando SVM y Redes Neuronales.....	67
9.4.2 Antes y después para STFT aplicando SVM y Redes neuronales	69
9.4.3 Antes y después para Wavelet aplicando SVM y Redes neuronales	71
10. CONCLUSIONES	74
11. TRABAJO FUTURO.....	75

ÍNDICE DE IMÁGENES.

Figura 1. Presentación de la banda Myo, donde se muestra la distribución de los sensores enumerados de 1 a 8 para los sensores de electromiografía y en donde en la posición 4 se encuentran el acelerómetro, magnetómetro y giroscopio [27].	20
Figura 2. Representación de señales continua y discreta donde A) representa la señal continua y B) representa la señal discreta [31].	21
Figura 3. Aplicación de la transformada de Fourier en una señal perteneciente a una señal donde A) es la representación de la señal en el dominio del tiempo y B) es la representación obtenida luego de aplicar la FFT a la señal continua(A).	23
Figura 4. Aplicación de la transformada Wavelet en una señal perteneciente a una señal donde A) es la representación de la señal en el dominio del tiempo y B) es la representación obtenida luego de aplicar la DWT a la señal continua(A).	25
Figura 5. Manera en la cual funciona la STFT.	26
Figura 6. Aplicación de la transformada corta de Fourier en una señal perteneciente a una señal donde A) es la representación de la señal en el dominio del tiempo y B) es la representación obtenida luego de aplicar la STFT a la señal continua(A).	26
Figura 7. Clasificación de las curvas según su curtosis, donde: curtosis>3 es leptocúrtica, curtosis=3 es mesocúrtica y curtosis<3 es platicúrtica [48].	28
Figura 8. Distribución simétrica según el sesgo obtenido [52].	28
Figura 9. Hiperplano de separación en espacio bidimensional de un conjunto [58].	30
Figura 10. Proceso de una red neuronal [59].	30
Figura 11. Resultados obtenidos representados en un diagrama de barras aplicando Relief a dos vectores X, Y donde se puede observar que la información con más relevancia son las primeras 5 columnas [61].	31
Figura 12. Representación gráfica del análisis de componentes donde las características son representadas en el eje X, la varianza en el eje Y donde se observa que 20 características son las que presentan la mayor varianza.	32
Figura 13. Matriz de correlación entre características donde los coeficientes varían de -0.5 a 1 representado por una escala de color donde si el color es muy amarillo quiere decir que está altamente relacionado.	32

Figura 14. Diagrama de bloques que muestra la metodología a seguir para llevar a cabo la clasificación de señas.....	33
Figura 15. Señas utilizadas para formar la base de datos [2].	33
Figura 16. Posicionamiento para la banda Myo [65].....	34
Figura 17. Estructura de la base de datos.	34
Figura 18. Interfaz gráfica de usuario o GUI donde se puede nombrar las señas tomas, el botón de “inicio” permite reconocer si hay conexión con la banda Myo y el botón “Start” inicia la toma de la muestra.	35
Figura 19. Diagrama de flujo para la toma de datos.....	35
Figura 20. Distribución de datos por sensor en cada muestra.	36
Figura 21. Estructura para la seña Bien.	36
Figura 22. Diagrama de flujo para aplicar la FFT donde el contador “i” indica el cambio entre estructuras y “j” el cambio entre muestras.	37
Figura 23. Estructura para las 12 señas comenzando desde la seña “Bien” hasta la seña “¿Qué paso?”.	37
Figura 24. Diagrama de flujo para aplicar DWT donde el contador “i” indica el cambio entre estructuras y “j” el cambio entre muestras.	38
Figura 25. Diagrama de flujo para la aplicación de la STFT donde el contador “i” indica el cambio entre estructuras y “j” el cambio entre muestras.....	39
Figura 26. Toma de características por canal	39
Figura 27. Diagrama de flujo para obtener las características de las estructuras de los descriptores tiempo-frecuencia.....	40
Figura 28. Estructura de los arreglos para el aprendizaje máquina para una seña.	40
Figura 29. Pasos a seguir para el entrenamiento con SVM donde: 1. Pestaña de APPS, 2. Classification Learner (Aprendizaje maquina), 3. New Session, 4. Workspace Variable (selección arreglo de características), 5. Start Session, 6. Selección de metodos de Machine Learning, 7. ALL SVMs (aplica diferentes kernel), 8. Train (Inicio de entrenamiento).	41
Figura 30. Arreglo binario pedido por la app de redes neuronales donde las 12 columnas indican las 12 señas y las 1200 filas indican la cantidad de muestras tomadas a todos los sujetos.	41

Figura 31. Pasos a seguir para el entrenamiento con Redes Neuronales 31 donde: 1. Pestaña de APPS, 2. Neuronal Net Pattern Recognition (Aprendizaje maquina), 3. Next, 4. Inputs (arreglo de características), 5. Targets (Arreglo binario), 6. Samples are (Especificar si las características son filas o columnas), 7. Training (Porcentaje del arreglo para entrenamiento), Validation (Porcentaje del arreglo para validacion), Testing (Porcentaje del arreglo para testeo), 8. Number of Hidden Neurons (número de capas), 9. Train (Entrenamiento del algoritmo).	42
Figura 32. Diagrama de flujo para la utilización de la función PCA.	43
Figura 33. Diagrama de flujo para la aplicación de la Correlación a el arreglo de características.....	43
Figura 34. Diagrama de flujo para la aplicación de Relieff a el arreglo de características. .	44
Figura 35. Arreglo ingresado con todos los sensores para en el entrenamiento con SVM. .	45
Figura 36. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 3 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	46
Figura 37. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 9 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2), entrega una mayor numero de aciertos.....	47
Figura 38. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 12 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3) y el quinto (1.5 Medium Gaussian SVM, Model 1.5), entregan una mayor numero de aciertos.	48
Figura 39. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 13 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	49
Figura 40. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 15 donde el quinto kernel aplicado (1.5 Medium Gaussian SVM, Model 1.5), entrega una mayor numero de aciertos.	50
Figura 41. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 3 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	51

Figura 42. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 6 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	52
Figura 43. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 13 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	53
Figura 44. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 14 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2), entrega una mayor numero de aciertos.....	54
Figura 45. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2) y el tercero (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	55
Figura 46. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 3 donde el tercero kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	56
Figura 47. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 8 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2), entrega una mayor numero de aciertos.....	57
Figura 48. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 11 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2) y el tercero (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	58
Figura 49. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 12 donde el tercero kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.....	59
Figura 50. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 15 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2), entrega una mayor numero de aciertos.....	60
Figura 51. Matriz de confusión obtenida de la app para la combinación 15 de la tabla1 usando FFT.	61
Figura 52. Matriz de confusión obtenida de la app para la combinación 15 de la tabla1 usando Wavelet.....	61

Figura 53. Matriz de confusión obtenida de la app para la combinación 15 de la tabla 1 usando STFT.....	62
Figura 54. Resultado aplicando PCA a el arreglo de características según la combinación 15 de la tabla 1.....	62
Figura 55. Resultado aplicando Relieff a el arreglo de características según la combinación 15 de la tabla 1.....	63
Figura 56. Resultado aplicando Correlación a el arreglo de características según la combinación 15 de la tabla 1.	63
Figura 57. Resultado aplicando PCA a el arreglo de características según la combinación 15 de la tabla 1.....	64
Figura 58. Resultado aplicando Relieff a el arreglo de características según la combinación 15 de la tabla 1.....	64
Figura 59. Resultado aplicando Correlación a el arreglo de características según la combinación 15 de la tabla 1.	65
Figura 60. Resultado aplicando PCA a el arreglo de características según la combinación 15 de la tabla 1.....	65
Figura 61. Resultado aplicando Relieff a el arreglo de características según la combinación 15 de la tabla 1.....	66
Figura 62. Resultado aplicando Correlación a el arreglo de características según la combinación 15 de la tabla 1.	66
Figura 63. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 40 (95.8%) y B son los nuevos resultados obtenidos (95.3%).	67
Figura 64. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 40 (95.8%) y B son los nuevos resultados obtenidos (89.7%).	68
Figura 65. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 51 (92.3%) y B son los nuevos resultados obtenidos (85.7%).	68

Figura 66. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 51 (92.3%) y B son los nuevos resultados obtenidos (72.8%).	69
Figura 67. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 50 (93.1%) y B son los nuevos resultados obtenidos (93.1%).	69
Figura 68. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 50 (93.1%) y B son los nuevos resultados obtenidos (62.5%).	70
Figura 69. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 53 (93.5%) y B son los nuevos resultados obtenidos (86.9%).	70
Figura 70. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 51 (93.5%) y B son los nuevos resultados obtenidos (47.2%).	71
Figura 71. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 45 (94.7%) y B son los nuevos resultados obtenidos (91.4%).	71
Figura 72. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 45 (94.7%) y B son los nuevos resultados obtenidos (90.6%).	72
Figura 73. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 52 (95.8%) y B son los nuevos resultados obtenidos (93.1%).	72
Figura 74. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 52 (95.8%) y B son los nuevos resultados obtenidos (88.3%).	73

RESUMEN

Este trabajo de carácter investigativo tiene como fin comparecer el análisis y resultados obtenidos a través de la aplicación de descriptores en frecuencia, caracterización de la información y aprendizaje maquina con el fin de clasificar 12 señas obtenidas del diccionario de lenguaje de señas colombiano. Dichos análisis y resultados servirán como precedente para implementación de un traductor de lenguaje de señas que permita a la comunidad sorda del país comunicarse con los no hablantes de el idioma.

La obtención de las señas se hace utilizando la banda Myo junto con el ambiente de desarrollo *Matlab*[®] el cual también es utilizado junto a varias de sus herramientas para la aplicación de los descriptores en frecuencia, características y aprendizaje máquina.

1. INTRODUCCIÓN

Para el año 2005 la cantidad de personas sordas en Colombia registradas por el DANE alcanzaba una cantidad mayor a las 500 mil, de las cuales aproximadamente un 35% eran analfabetas [1]. Actualmente aún no se tiene un dato exacto de cuantas personas son sordas, pero si se tiene claro que gracias a las estrategias que se están llevando para impartir el lenguaje de señas colombiano (LSC) [2] por entidades como la Federación Nacional de Sordos de Colombia (FENASCOL) [3] o gubernamentales como INSOR [4] y distintas universidades [5], el número de personas sordas que pueden comunicarse es mayor.

Según un artículo publicado el 21 de febrero de 2018 por EL TIEMPO y con ayuda de INSOR se dice que, el lenguaje de señas es la segunda lengua más hablada en Colombia, aclarando que no solo hay un lenguaje de señas, *“Si bien hay unos símbolos universales, esta forma de comunicación se adapta a las necesidades de cada región del mundo. No hay ninguna exactamente igual. De hecho, en nuestro país se le llama ‘Lengua de señas colombiana’ y varía dependiendo de la región donde se hable. Por ejemplo, en Antioquia hay una señal especial para decir el tan común ‘eh, ave maría’.*

Pero no solo eso. Al igual que el español hablado, la entonación cambia de acuerdo a la región. En la costa, la lengua de señas se hace con los brazos más abiertos y movimientos mucho más ‘cadenciosos’. Por el contrario, en las ciudades de la región andina, los movimientos son más recatados. Las manos y los brazos permanecen más pegados al cuerpo, explican los especialistas del Insor.” [6].

De acuerdo con los resultados arrojados por el DANE en el censo de 2018 el cual cuantifica las personas en el país y el lugar donde se encuentran, estima que en Colombia para dicho año hay 48.258.494 personas [7], dicho esto, si comparamos esta cantidad de personas en Colombia y las 500 mil personas sordas registradas en el año 2005, se puede decir que la mayoría de personas en Colombia no podrían sostener una conversación bidireccional a través del lenguaje de señas colombiano, debido a que este no es impartido como segunda lengua en todo el país.

2. PLANTEAMIENTO DEL PROBLEMA

Partiendo del hecho de que en Colombia la mayoría de ciudadanos no tienen conocimiento sobre el LSC ni lo practican, se crea una brecha lingüística que mantiene la comunidad sorda aislada, además, otros componentes como los dialectos del lenguaje de señas encontrados en distintas partes del país hacen crecer dicha brecha. Una persona sorda se enfrenta a diferentes retos en su vida cotidiana referentes a la manera en que se comunican, como lo pueden ser en la asistencia del transporte urbano, la información sobre direcciones o lugares, compra de elementos para la vida diaria entre las muchas otras situaciones que se presentan en el día a día.

Hasta el momento, para dar solución a los problemas de quienes solo pueden comunicarse a través del lenguaje de señas, se presentan propuestas como la implementación de intérpretes en puntos específicos de sectores como la televisión [8], además se brindan herramientas como el centro de relevos [9] que propina herramientas como el relevo de llamadas que permite comunicación bidireccional entre personas sordas y oyentes durante rangos de tiempo, sistema de interpretación en línea, aprendizaje a través de contenido en lenguaje de señas y una aplicación móvil para la comunicación vía chat o video, sin embargo, a pesar de dicho esfuerzo, sigue siendo evidente que la comunidad sorda no logra interactuar de manera activa en la sociedad y siguen estando limitados en ciertas zonas o ámbitos [6].

Hay que decir que la comunidad sorda tiene dos opciones para comunicarse con otros, la manera escrita y utilizando lenguaje de señas. Hablando de manera práctica, la comunicación escrita tiene distintos requerimientos para ser llevada a cabo, como el lugar, ambiente y materiales más que suficientes, por lo tanto, no es la manera más viable en todo momento; lo que deja el lenguaje de señas, el cual es más dinámico y versátil, por ende, el lenguaje de señas es la única lengua que muchas personas de la comunidad sorda hablan. Con esto, es correcto seguir una línea de pensamiento donde, si se quiere lograr una mayor inclusión social para la comunidad sorda, más ciudadanos colombianos tengan conocimientos sobre el lenguaje de señas o tengan forma de entenderlo.

Entonces, ¿Cómo lograr que un mayor número de personas que no manejan el Lenguaje de Señas Colombiano puedan comunicarse con quienes si lo usan como lengua nativa?

Teniendo en cuenta que Colombia supera los 48 millones de habitantes la idea de implementar estrategias para impartir el lenguaje de señas colombiano a quienes no lo conocen, sería un reto difícil de superar, por tanto, es necesario buscar otras alternativas y para esto es necesario observar el panorama mundial y nacional.

Según la Organización Mundial de la salud 466 millones de personas en todo el mundo padecen pérdida de audición y se cree que para el 2050 por cada 10 personas 1 sufrirá una pérdida de audición; de acuerdo con esto en pro de mejorar la situación de las personas con pérdida de audición se implementan la utilización de audífonos, implantes cocleares y dispositivos de ayuda, implementación de subtítulos y la adopción del lenguaje de signos [10]. Con este aumento de la población sorda también comienzan a emerger herramientas e investigaciones desarrolladas a nivel mundial que se centran en interpretar el lenguaje de señas para las personas quienes no lo conocen, como las IA (Inteligencia Artificial) desarrolladas por Google para la interpretación de lenguaje de señas británico a lenguaje escrito, en el cual no ofrecen el traductor si no algoritmos más robustos para el seguimiento de las señas y la conversación bilateral [11], también se crean sistemas de traducción como el encontrado en *“Real-Time Malaysian Sign Language Translation using Colour Segmentation and Neural Network”* que traduce el lenguaje de señas de Malasia en tiempo real captado en video [12] y diferentes sistemas de interpretación como **“Sistema traductor de lenguaje de señas mexicanas”** que utiliza la visión por computador usando el sistema Android [13]. Muchas de estas alternativas han sido desarrolladas en Colombia como las encontradas en *“hetah.net”* [14] las cuales consisten en un traductor en línea representado por un avatar 3d, el cual interpreta en lenguaje de señas la oración ingresada, y una aplicación para celular donde a través de vibraciones traduce el texto escrito en código Morse, también se encuentran investigaciones como **“Clasificación automática de las vocales en el lenguaje de señas colombiano”** donde a través del procesamiento digital de imágenes reconocen las letras del abecedario [15], **“Prototipo traductor de señales manuales a texto legible, utilizando Kinect”** donde utilizan el procesamiento digital de imágenes junto con la tecnología Kinect para lograr la identificación de señas más complejas [16], **“Diseño e implementación de un algoritmo para interpretar las vocales del lenguaje de señas colombiano a través de sensores”** en donde de manera alternativa se usan sensores junto al procesamiento digital de señal para identificar las señas pertenecientes al abecedario [17],

“Reconocimiento del Lenguaje de Señas Manuales con el Kinect” donde utilizan la tecnología Kinect y el procesamiento digital de imágenes para caracterizar señas más complejas [18].

A partir de las anteriores investigaciones y desarrollos vistos se debe decir que el problema radica en que tomar como enfoque el abecedario debido a que esto solo tiene utilidad en algunos temas y no es utilizado para formar una conversación fluida, por otro lado, es necesario el análisis de todos los queremas¹ que dan significado a cada seña y al mismo tiempo que todo esto pueda ser aplicado a la vida diaria. Todo esto con el fin de que una persona pueda formar oraciones que puedan ser llevadas a quienes no conocen el lenguaje de señas colombiano.

¹ El termino queremas hace referencia a las diferentes partes que dan significado al signo como posición de la mano, orientación, lugar, expresión corporal entre otros. Es decir, es equivalente a los fonemas en el lenguaje hablado [19] [20].

3. JUSTIFICACIÓN

A pesar de que existen herramientas que intentan cerrar la brecha lingüística entre quienes usan el lenguaje de señas colombiano y quienes no tienen conocimiento de dicha lengua, no se toman algunos aspectos relevantes alusivos a las necesidades del usuario como lo es libre movimiento de forma cómoda y a su vez la transmisión efectiva de la información; con base en esto, se presenta una opción que tiene en cuenta estos aspectos utilizando las prestaciones dadas por la banda Myo [21] como son los distintos tipos de sensores que esta incluye que son acelerómetro, magnetómetro, giroscopio y sensores de electromiografía, los cuales cuentan con una gran protección contra ruido y gran sensibilidad frente a los movimientos hechos por todo el brazo y dedos, además que esta se comunica por Bluetooth.

Con esta banda se espera lograr una caracterización del lenguaje de señas que permita captar señas más complejas, y para dar apoyo a este argumento se toman diferentes trabajos de investigación encontrados en Google Académico y en bases de datos sobre investigaciones como la *IEEE Xplore® Digital Library*, con la cual se utilizan los sensores de electromiografía, para la caracterización de señales de los diferentes lenguajes de señas a nivel mundial. Este es el caso de una investigación publicada en 2016 que se encuentra en la base de datos de la IEEE llamada **“Evaluating Sign Language Recognition Using the Myo Armband”** [22] donde a través del sensor electromiográfico (EMG) que contiene la banda Myo, logran desarrollar un sistema de reconocimiento de las letras del abecedario del lenguaje de señas brasileño, programando un sistema de Machine Learning conocido como SVM o máquina de soporte vectorial con lo que lograron el reconocimiento de las señas correspondientes, por el mismo estilo se encuentran documentos como **“Gesture Recognition for Indonesian Sign Language Systems (ISLS) Using Multimodal Sensor Leap Motion and Myo Armband Controllers Based-on Naïve Bayes Classifier”** [23] en el cual utilizando una banda Myo y un sensor Leap Motion, se hace el reconocimiento de algunas letras del abecedario indonesio, basados en un clasificador estadístico conocido como “Naïve Bayes”.

4. OBJETIVO GENERAL

Realizar un análisis de características tiempo frecuencia para la clasificación de lenguaje de señas colombiano utilizando sensores invasivos

5. OBJETIVOS ESPECÍFICOS

- Realizar la captura de una base de datos de 10 señas simétricas del lenguaje de señas colombianas por medio de la banda Myo
- Implementar descriptores tiempo frecuencia que permitan generar un espacio vectorial de representación.
- Implementar algoritmos de análisis de datos con el fin de determinar el conjunto de características que representen mejor los datos.
- Implementar y validar dos algoritmos de aprendizaje de máquina.

6. ESTADO DEL ARTE

Partiendo del hecho de que la comunidad sorda no solo se limita al estado colombiano, sino a todo el mundo, se logra identificar que diferentes personas han hecho estudios e investigaciones en pro de dar herramientas para que el lenguaje de señas pueda ser entendido por una mayor población. Dentro de estos estudios se pueden encontrar distintas alternativas con respecto a métodos y tecnologías usadas, como se puede apreciar en documentos alojados en IEEE Xplore[®] Digital Library como: ***“Deaf talk using 3D animated sign language: A sign language interpreter using Microsoft's kinect v2”*** [24] , provisto desde la National University of Science and technology (NUST) en Pakistán en el cual se utilizan herramientas suministradas por Microsoft para el dispositivo “Kinect” donde se toman oraciones en lenguaje hablado y se convierten a través de un avatar 3d en las señas correspondientes, ***“American Sign Language Recognition using Deep Learning and Computer Vision”*** [25] es otra investigación proveniente desde Kennesaw State University en Estados Unidos, en donde se utilizan algoritmos de visión por computador y se clasifican las señas captadas en video a través de un celular de alta gama, ***“Hand Gesture Recognition for Deaf People Interfacing”*** [26] proveniente desde España, en donde se utiliza procesamiento digital de imágenes para reconocer las señas del abecedario.

Los artículos mencionados anteriormente desarrollados en distintos países son solo algunos de los tantos que se muestran en la red, los cuales permiten la clasificación del lenguaje de señas o la traducción desde el lenguaje escrito o hablado a el lenguaje de señas. Estos a su vez dejan ver hasta qué punto ha sido tratado este problema, además evidencia cuáles son las tecnologías o métodos usados hasta el momento fuera de Latinoamérica, entre estos cabe resaltar que las herramientas más utilizadas entre las muchas investigaciones encontradas son dispositivos que permitan la captura de movimiento como cámaras o el ya mencionado “Kinect” además de los algoritmos de aprendizaje máquina para la clasificación de estas señas.

Prestando atención a las investigaciones en Latinoamérica como, ***“Real-Time Mexican Sign Language Recognition”*** [27] la cual utiliza el dispositivo “Kinect” para reconocer en tiempo real las señas pertenecientes a el abecedario, ***“Evaluating a Speech Communication System for Deaf People”*** [28], que permite llevar el lenguaje hablado a el lenguaje de señas por

medio de un avatar 3d; se evidencia la utilización de dispositivos y métodos parecidos para el reconociendo o traducción de señas.

Profundizando a nivel nacional, en Colombia también se encuentran distintas investigaciones abordadas de diferentes maneras, para brindar herramientas a la comunidad sorda como ***“Application Design Sign Language Colombian for Mobile Devices”*** [29] propuesta desde la Universidad Santo Tomas, la cual trata sobre un aplicativo móvil el cual convierte de lenguaje hablado a lenguaje de señas, ***“Application to support the process of training sign language through Microsoft Kinect®”*** [30] propuesta desde la Universidad Piloto de Colombia, donde se utiliza procesamiento digital de imágenes para reconocer las señas para distintos ámbitos, ***“DESARROLLO DE UN PROTOTIPO DE SOFTWARE COMO HERRAMIENTA PARA LA ENSEÑANZA DEL LENGUAJE DE SEÑAS COLOMBIANAS”*** [31] propuesta desde la Universidad de San Buenaventura, que a través del procesamiento de imágenes permite identificar las letras del abecedario.

7. MARCO TEÓRICO

7.1 Sensores

Puesto que se busca la utilización de la banda Myo para la clasificación de diferentes señas, se parte de una definición que se ajusta para este caso de lo que es un sensor.

“Un sensor es un dispositivo eléctrico y/o mecánico que convierte magnitudes físicas (luz, magnetismo, presión, etc.) en valores medibles de dicha magnitud. Esto se realiza en tres fases:

- Un fenómeno físico a ser medido es captado por un sensor, y muestra en su salida una señal eléctrica dependiente del valor de la variable física.*
- La señal eléctrica es modificada por un sistema de acondicionamiento de señal, cuya salida es un voltaje.*
- El sensor dispone de una circuitería que transforma y/o amplifica la tensión de salida, la cual pasa a un conversor A/D, conectado a un PC. El convertidor A/D transforma la señal de tensión continua en una señal discreta.”² [33].*

Se ha de tener en cuenta que la banda Myo cuenta con 4 tipos de sensores diferentes, acelerómetro, magnetómetro, de electromiografía y giroscopio los cuales se distribuyen por toda la banda como se muestra en la figura 1.



Figura 1. Presentación de la banda Myo, donde se muestra la distribución de los sensores enumerados de 1 a 8 para los sensores de electromiografía y en donde en la posición 4 se encuentran el acelerómetro, magnetómetro y giroscopio [34].

Teniendo en cuenta todas las posibles aplicaciones evidenciadas en *“Technical Features and Functionalities of Myo Armband: An Overview on Related Literature and Advanced Applications of Myoelectric Armbands Mainly Focused on Arm Prostheses”* [35] que a partir de gestos permiten la interacción con dispositivos como televisores, drones, robots,

² La utilización de las siglas “A/D” hacen referencia a un convertidor análogo digital el cual permite llevar una señal de tiempo continuo a tiempo discreto para que dispositivo de procesamiento como un computador u otro dispositivo pueda captar la información. [32].

sistemas médicos, y distintos tipos de prótesis, surge una pregunta y es ¿Cómo es posible lograr estas cosas con la banda Myo y cómo puede llevarse esto a la clasificación de señas del lenguaje de señas colombiano?.

Para responder a esta pregunta es necesario explicar el concepto de **Procesamiento digital de señal**³ (PDS), el cual se encarga de tomar la información en una señal discreta y transformarla o manipularla a través de procesos matemáticos [37]. Una señal discreta se puede tomar como una señal formada por muestras que son tomadas cada determinado tiempo de una señal continua, esto se debe a que cada proceso dependiendo del dispositivo, demanda un tiempo mínimo para ser llevado a cabo, en la figura 2 se observa la representación de una señal continua y discreta en el dominio del tiempo.

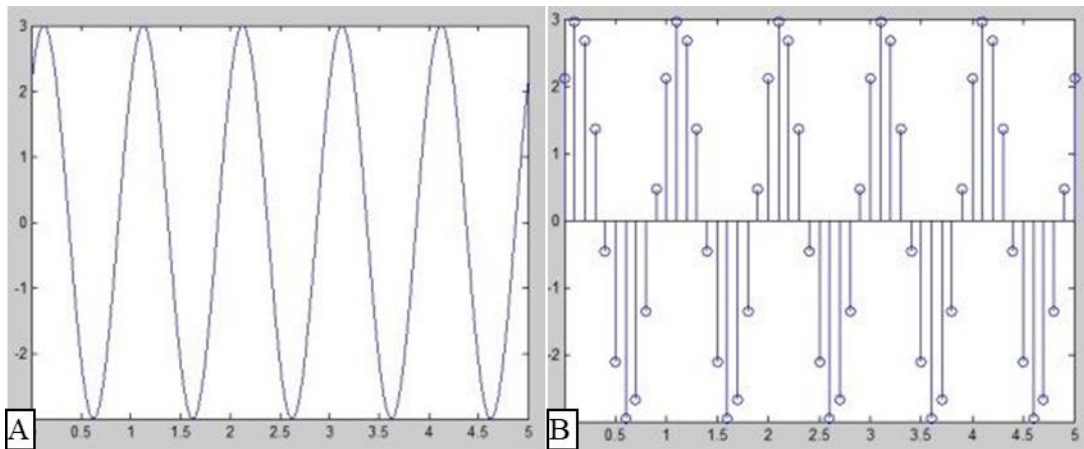


Figura 2. Representación de señales continua y discreta donde A) representa la señal continua y B) representa la señal discreta [38].

De acuerdo con lo anterior lo entregado por la banda Myo son señales discretas provenientes de cada sensor, a continuación, se da una breve explicación sobre el funcionamiento de cada sensor.

7.1.1 Sensor de electromiografía

Este tipo de sensor es muy frecuente en dispositivos clínicos o terapéuticos debido a que este permite detectar y analizar la actividad muscular, por otro lado, se ha de tener en cuenta que el tipo de señal entregada por estos sensores es tomada como pequeña señal debido a que las tensiones obtenidas por la contracción muscular varían a rededor de 100mv [39].

7.1.2 Unidad de medición inercial (IMU):

Es un dispositivo el cual permite conocer posición, orientación o fuerzas gravitacionales y velocidad [40][41], está compuesto por 3 tipos de sensores:

Magnetómetro:

³ En electrónica una señal (eléctrica) es tomada como aquella que provee información y se ha de tener en cuenta que dichas señales pueden ser análogas o digitales las cuales tienen distintas características como en la amplitud [36]

Este sensor es el que permite conocer la orientación o fuerza gravitacional en el dispositivo debido a que su respuesta depende del campo magnético terrestre y seguía por el norte terrestre.

Acelerómetro:

Permite conocer la aceleración ya que responde de acuerdo al movimiento ejercido.

Giroscopio:

Responde respecto a los giros realizados al mover la banda.

7.2 Descriptores Tiempo-Frecuencia

Debido a que cada señal tomada está en el dominio del tiempo se hace necesario la utilización de descriptores en frecuencia que permitan facilitar el análisis y poder capturar la información de interés, para esto se tienen 3 opciones diferentes de descriptores en frecuencia, Transformada rápida de Fourier cuyas siglas en ingles son FFT (*Fast Fourier Transform*), Transformada Wavelet y Transformada corta de Fourier cuyas siglas en ingles son STFT (*Short-Time Fourier Transform*) [42] las cuales se explican a continuación:

7.2.1 Transformada rápida de Fourier (FFT)

Remontando a lo dicho Joseph Fourier, que toda función periódica de periodo T puede escribirse como la suma de funciones seno y coseno con lo cual se puede presentar como la suma de una serie de Fourier cuyos coeficientes están dados por:

$$a_k = \int_{t_0}^{t_0+T} f(x) \cos 2\pi \omega_k x \, dx \quad b_k = \int_{t_0}^{t_0+T} f(x) \sin 2\pi \omega_k x \, dx$$

Donde t_0 es un punto cualquiera, ω_k es la frecuencia angular y $f(x)$ representa la función [43].

Teniendo en cuenta que la anterior forma de aplicar la transformada de Fourier es para señales continuas se utiliza la ecuación 1 y 2⁴ puesto que las señales que provee la banda Myo son discretas.

$$y(k) = \sum_{j=1}^n X(j)W_n^{(j-1)(k-1)} \quad (1)$$

$$X(j) = \frac{1}{n} \sum_{k=1}^n Y(k)W_n^{-(j-1)(k-1)} \quad (2)$$

Donde $W_n = e^{(-2\pi i)/n}$ y n es la longitud del vector por otro lado j y k indican posiciones espaciales dentro del vector de muestras de la señal.

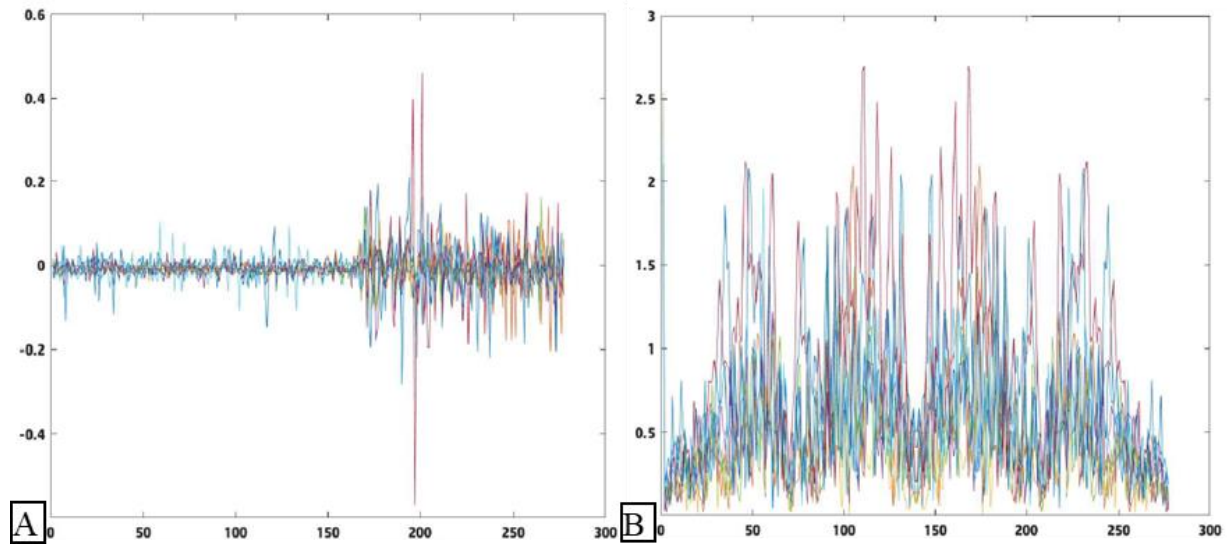


Figura 3. Aplicación de la transformada de Fourier en una señal perteneciente a una seña donde A) es la representación de la seña en el dominio del tiempo y B) es la representación obtenida luego de aplicar la FFT a la señal continua(A).

⁴ Las ecuaciones 1 y 2 fueron tomadas de la documentación provista por MathWorks® [44] puesto que son las utilizadas por el entorno de desarrollo *Matlab*®.

7.2.2 Transformada Wavelet

También conocida ondelettes u onditas son funciones que permiten la aproximación de datos con variaciones o con discontinuidades abruptas, la base teórica de esta proviene de la teoría de Fourier y son utilizadas entre otras cosas para el procesamiento digital de señal [45]. Se debe aclarar que este tipo de funciones no toma señales senos y cosenos para la aproximación, sino más bien toma pequeñas ondas [46] donde a partir de la dilatación y traslación de la onda base (madre [47]) se creara una familia que al ser sumada generara la aproximación, a continuación, se observa como de manera similar que la FFT se aplica la transformada wavelet continua.

$$WT(b, a) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t) \psi\left(\frac{(t - b)}{a}\right) dt$$

Donde $x(t)$ es la señal, $\psi(t)$ es la función wavelet base (madre), “a” es el parámetro de dilatación y “b” es el parámetro de traslación.

De la misma forma que en la FFT cuando se pasa del tiempo continuo al tiempo discreto la función y la aplicación de la wavelet cambia, además, se debe tener en cuenta que la onda madre que se usa para esta transformada wavelet discreta (DWT) debe tener energía unitaria por tanto la familia de wavelets estaría dada por la ecuación 3.

$$\psi_{j,k}(t) = 2^{\frac{-j}{2}} \psi(2^{-j}t - k) \quad (3)$$

En donde j indica el ancho de la wavelet y k la posición de esa forma se dilata y traslada la función madre $\psi(t)$.

Para el análisis de datos en distintas resoluciones es necesario usar la función de escalamiento vista en la ecuación 4.

$$\phi(t) = \sum_{k=-1}^{N-2} (-1)^k \cdot C_{k+1} \psi(2t + k) \quad (4)$$

Donde C_k puede ser tomado como un filtro el cual consiste en una matriz de transformación con sus diferentes coeficientes en dos patrones los cuales funcionan como un filtro pasa-bajo y pasa-alto⁵.

⁵ Las ecuaciones 3 y 4 y la información tomada sobre la transformada wavelet fueron tomadas de la referencia [45] donde se explica mejor, de igual forma en [46] y [47] se complementa dicha información.

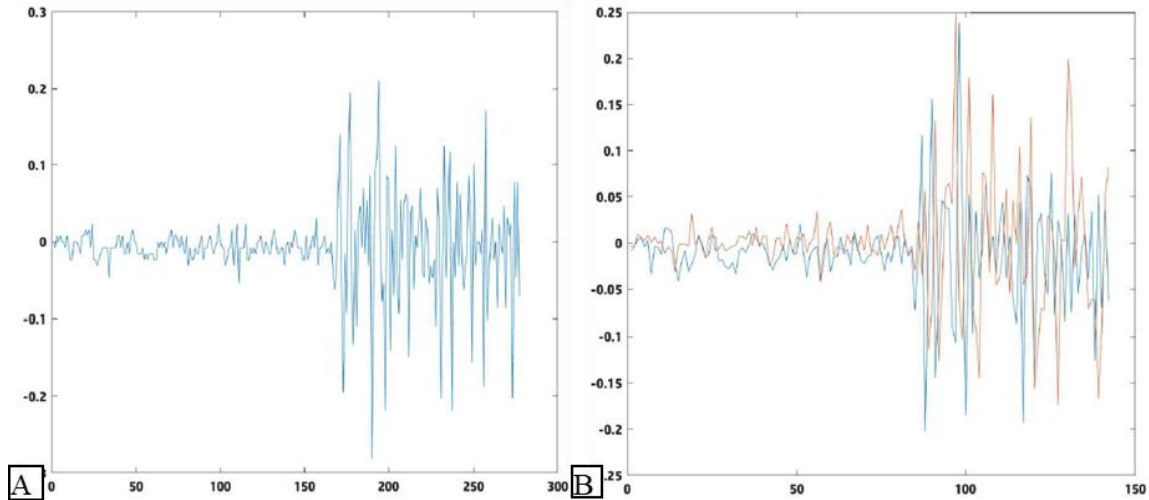


Figura 4. Aplicación de la transformada Wavelet en una señal perteneciente a una seña donde A) es la representación de la seña en el dominio del tiempo y B) es la representación obtenida luego de aplicar la DWT a la señal continua(A).

7.2.3 Transformada corta de Fourier

Es también conocida como la Transformada de Fourier “ventaneada” [48] y al igual que la FFT es utilizada comúnmente para el procesamiento digital de señal, pero a diferencia de esta la manera en cómo funciona la STFT es dividiendo la señal en varios segmentos a los cuales le aplica la Transformada de Fourier esto permite no solo conocer los componentes en frecuencia de la señal si no también la ubicación en el tiempo.

Debido a que la STFT [49] se aplica en una función discreta se deben tener en cuenta las divisiones de la señal o “ventanas” para aplicar la Transformada de Fourier (TF) como se muestra en la ecuación 5 y 6.

$$k = \left\lfloor \frac{N_x - L}{M - L} \right\rfloor \quad (5)$$

Donde k hace referencia a la cantidad de ventanas en las que se aplicara la TF, N_x sera la cantidad de muestras de la señal original, M sera el tamaño de cada ventana y L es un parámetro que compensa la perdida de información al cambiar de cada ventana como se muestra en la figura 5.

$$X_m(f) = \sum_{n=-\infty}^{\infty} x(n)g(n - mR)e^{-j2\pi fn} \quad (6)$$

Donde X_m será la Transformada de Fourier Discreta en “ventana”, $x(n)$ la señal original, $g(n)$ es la ventana de tamaño M y R es el tamaño que debe saltar entre datos para ubicar las ventanas.

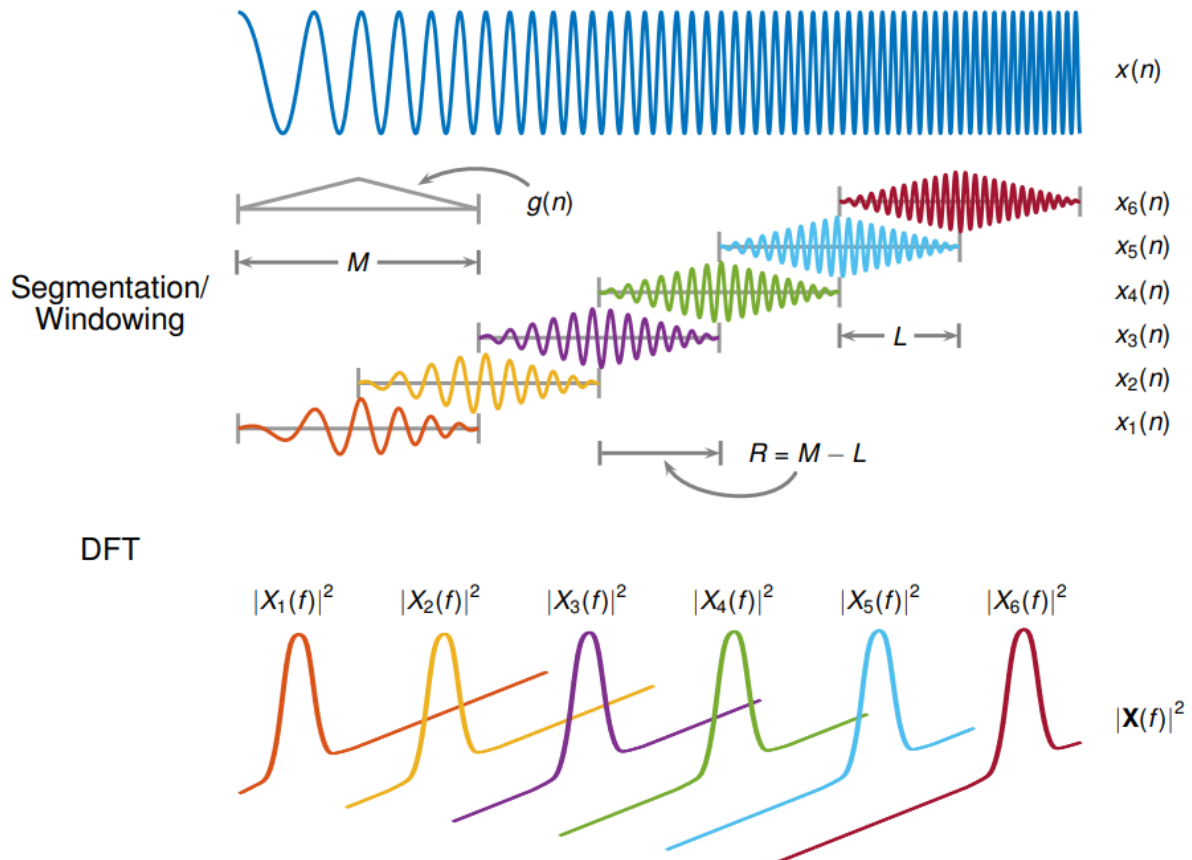


Figura 5. Manera en la cual función la STFT.

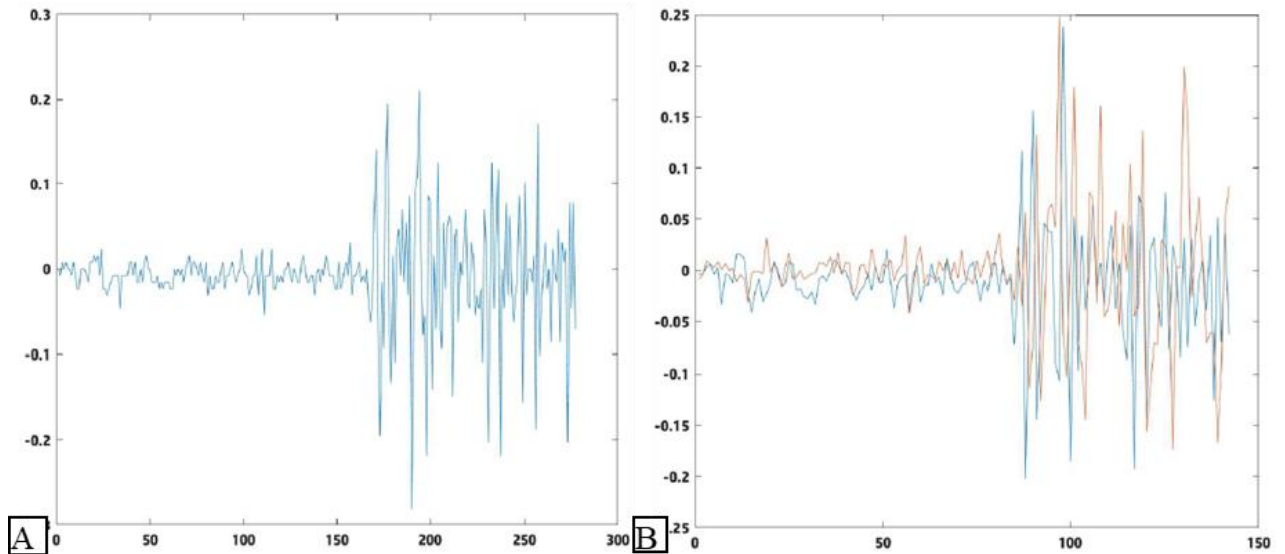


Figura 6. Aplicación de la transformada corta de Fourier en una señal perteneciente a una señal donde A) es la representación de la señal en el dominio del tiempo y B) es la representación obtenida luego de aplicar la STFT a la señal continua(A).

7.3 Características

Debido a que es necesario la utilización de algunas características para el procesamiento digital de señal se tiene en cuenta las siguientes:

7.3.1 Media

También conocido como promedio [50] con el cual se puede representar un grupo de datos obtenidos del mismo fenómeno que presentan diferencias en sus valores, para calcularlo se utiliza la ecuación 7⁶.

$$\mu = \frac{1}{n} \sum_{i=1}^n A_i \quad (7)$$

donde μ es el valor promedio, n son el numero de datos y A representa cada uno de los valores que se usan para el promedio.

7.3.2 Desviación estándar

Tomando en cuenta un grupo de datos que presentan diferentes valores entre sí, la desviación estándar indica que tan dispersos están los datos con respecto a la media de los mismos [52], para calcularla se utiliza la ecuación 8⁷.

$$S = \sqrt{\frac{1}{N-1} \sum_{i=1}^n |A_i - \mu|^2} \quad (8)$$

Donde S es la desviación estándar, N es el número de datos, A representa cada uno de los valores que se usan y μ es el valor promedio del grupo de datos.

7.3.3 Curtosis

Esta se basa en una curva tipo campana que permite saber por medio de la forma de la misma si los datos son cercanos a la media del grupo de datos, es decir indica si hay mayor o menor distribución de los datos [54].

Se considera que para una distribución normal de datos su curtosis tiene un valor de 3, es decir los datos tienen una dispersión cercana a la media, pero cuando dicha dispersión cambia ya sea mayor o menor a 3 nos indica que la distribución no es normal por tanto se manejan 3 tipos de curvas como se muestra en la figura 7.

⁶ La ecuación 7 para calcular el valor promedio es tomada de la documentación provista por MathWorks® [51] puesto que son las utilizadas por el entorno de desarrollo *Matlab*®.

⁷ La ecuación 8 para calcular la desviación estándar es tomada de la documentación provista por MathWorks® [53] puesto que son las utilizadas por el entorno de desarrollo *Matlab*®.

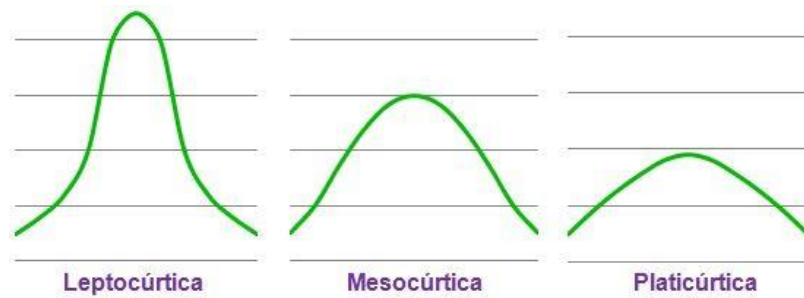


Figura 7. Clasificación de las curvas según su curtosis, donde: curtosis>3 es leptocúrtica, curtosis=3 es mesocúrtica y curtosis<3 es platicúrtica [55].

En caso de que la curtosis sea del tipo leptocúrtica indica que una gran cantidad de datos se acercan a la media, mesocúrtica una cantidad normal de datos se acercan a la media y platicúrtica indica que muchos de los datos no están cercanos a la media, en la ecuación 9⁸ se indica como se aplica la curtosis.

$$k_1 = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^4}{\left(\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \right)^2} \quad (9)$$

Donde k_1 es el valor de curtosis calculado, \bar{x} indica el valor promedio, n la cantidad de datos y x_i representa cada uno de los datos utilizados.

7.3.4 Sesgo

Al igual que la curtosis la base la el sesgo es una curva en forma de campana, pero a diferencia de la curtosis esta tiene que ver más con la deformación de la curva y esto indicara un error [57]. La forma en cómo se interpreta el sesgo está relacionado a la simetría de la curva como se observa en la figura 8 donde se evidencia un sesgo positivo que indica que la mayoría de los datos usados tienen un valor por encima del promedio y un sesgo negativo que indica que la mayoría de los datos usados tienen un valor por debajo del promedio [58]. Para aplicar el sesgo se utiliza la ecuación 10⁹.



Figura 8. Distribución simétrica según el sesgo obtenido [60].

⁸ La ecuación 9 para calcular la curtosis es tomada de la documentación provista por MathWorks® [56] puesto que son las utilizadas por el entorno de desarrollo *Matlab*®.

⁹ La ecuación 10 para calcular el sesgo es tomada de la documentación provista por MathWorks® [59] puesto que son las utilizadas por el entorno de desarrollo *Matlab*®.

$$s_1 = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^3}{\left(\sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \right)^3} \quad (10)$$

Donde s_1 es el valor de sesgo calculado, \bar{x} indica el valor promedio, n la cantidad de datos y x_i representa cada uno de los datos utilizados.

7.4 Aprendizaje máquina (Machine Learning)

Juega un papel importante en la actualidad en temas como la minería de datos, procesamiento digital de señal e imágenes, clasificación de información entre muchas otras aplicaciones. El aprendizaje máquina se puede definir como un campo en el cual se le brinda a la computadora la capacidad de aprender, esto se logra a través de algoritmos los cuales son entrenados con datos del fenómeno para el que se aplica, permitiendo así que la computadora tome decisiones [61][62].

Existen dos tipos de aprendizaje máquina, **supervisado** y **no supervisado**, para el primero se le entregan los datos para el entrenamiento los cuales consisten en características y etiquetas donde las características pueden ser tomadas como preguntas y las etiquetas como las respuestas a dichas preguntas, de esta manera el algoritmo será predictivo con respecto a las características ingresadas y para el segundo caso solo se le entregaran características así el algoritmo solo se concentrara en encontrar similitudes y en la agrupación [63].

7.4.1 Máquina de soporte vectorial (SVM)

Las SVM (*Support Vector Machines*) están desarrolladas sobre el aprendizaje de manera estadística y aunque fueron pensados originalmente para resolver problemas de clasificación binaria en la actualidad se utilizan en muchos otros aspectos y se utilizan en campos como el procesamiento digital de señal, visión por computador y categorización de texto entre otros.

Las SVM son clasificadores lineales y es uno de los muchos métodos usados para el aprendizaje máquina de forma supervisada, gracias a los denominados “kernel” [64] es posible llevar a cabo métodos de aprendizaje, puesto que dichos “kernel” son funciones matemáticas que permiten convertir los problemas de clasificación no lineales en espacios lineales con mayores dimensiones como lo sería un espacio de características. En la figura 9 se muestra de manera gráfica como es hecha una clasificación o separación en un espacio bidimensional de un conjunto donde se define un hiperplano¹⁰ como una función lineal.

¹⁰ Un hiperplano es una extensión del concepto de plano que tiene como fin dividir el espacio dimensional es decir en un plano unidimensional un hiperplano es el punto que divide una línea recta en dos semi-líneas rectas o hablando de un plano bidimensional el hiperplano es una línea recta que divide el plano en dos planos [65].

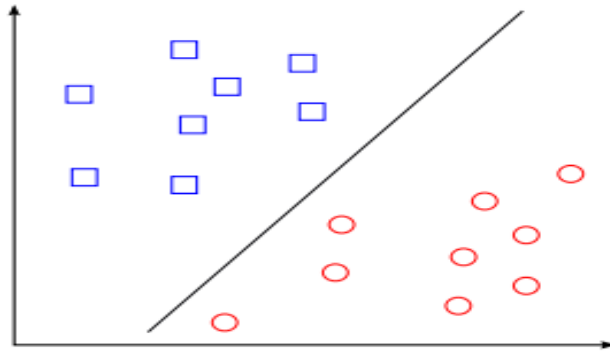


Figura 9. Hiperplano de separación en espacio bidimensional de un conjunto [66].

7.4.2 Redes Neuronales

Al igual que el sistema celular que compone nuestro cerebro (neuronas) las cuales están interconectadas entre sí, con el paradigma de las redes neuronales artificiales se busca recrear esta estructura donde cada célula procesa información en algunos casos enviada desde otras neuronas o directamente desde el exterior. Para la recreación de esta red se debe tener en cuenta aspectos como el establecimiento de nuevas conexiones entre neuronas, la ruptura en la conexión, la recreación de nuevas neuronas cuando otras mueran, además a nivel de procesamiento se debe tener en cuenta el procesamiento en paralelo puesto que en el cerebro miles de millones de neuronas trabajan en paralelo en procesos como la visión, memoria distribuida para que no exista perdida de información en caso tal de que una neurona muera y por último se debe adaptar al entorno para que pueda aprender de diferentes casos y pueda generar conclusiones como se muestra de manera gráfica en la figura 10 [67].

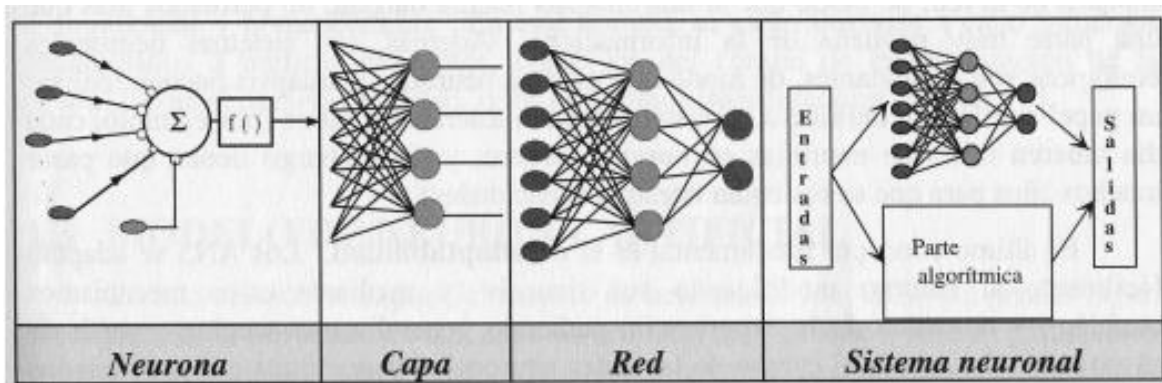


Figura 10. Proceso de una red neuronal [67].

Retomando la figura 10 se dice que la neurona es el elemento básico para un sistema neuronal las cuales están agrupadas entre sí y forman diferentes capas que a su vez forman lo que se conoce como la red neuronal, agregando las interfaces de entrada y salida entonces se obtiene el sistema neuronal, por otro lado, cada neurona tiene un algoritmo que puede cambiar entre capas debido a que se utilizan diferentes neuronas [67].

7.5 Análisis de componentes

Retomando el concepto del aprendizaje maquina supervisado el cual se enfoca en predecir una respuesta de acuerdo a sus características, se debe tener en cuenta que debido a la gran cantidad de información no se sabe cuanta de esta es relevante para el clasificador es decir que puede existir información que no aporte, por tanto, es necesario analizar dicha información y para esto se presentan los siguientes métodos.

7.5.1 Relieff

La manera en cómo este método clasifica la información es a través de pesos, es decir a través de diferentes algoritmos les asigna un peso a las diferentes características referenciando en los vecinos cercanos, mostrando que características son irrelevantes y cuales funcionan bien en grupo, aunque no muestra que información es redundante [68] como se muestra en la figura 11.

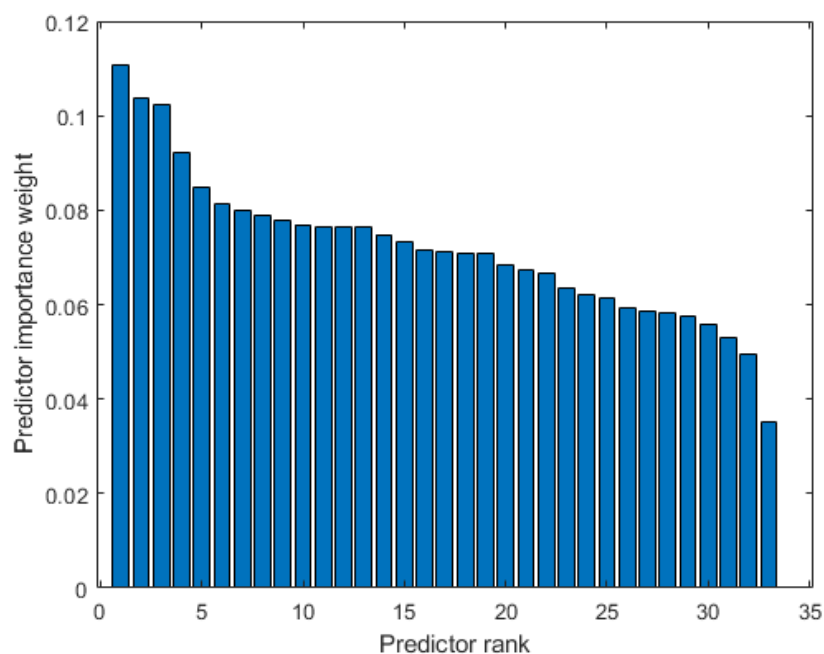


Figura 11. Resultados obtenidos representados en un diagrama de barras aplicando Relieff a dos vectores X, Y donde se puede observar que la información con más relevancia son las primeras 5 columnas [69].

7.5.2 PCA (Principal Component Analysis)

Este método está basado en la varianza¹¹ en donde el algoritmo donde encuentra los componentes que expliquen las mayores cantidades de varianza y organizándolos por tanto

¹¹ La varianza se refiere a que tanto varia una serie de datos respecto a la media de los mismos [70].

se puede tomar como un clasificador no supervisado [68], en la figura 12 se muestra un ejemplo gráfico de su funcionamiento.

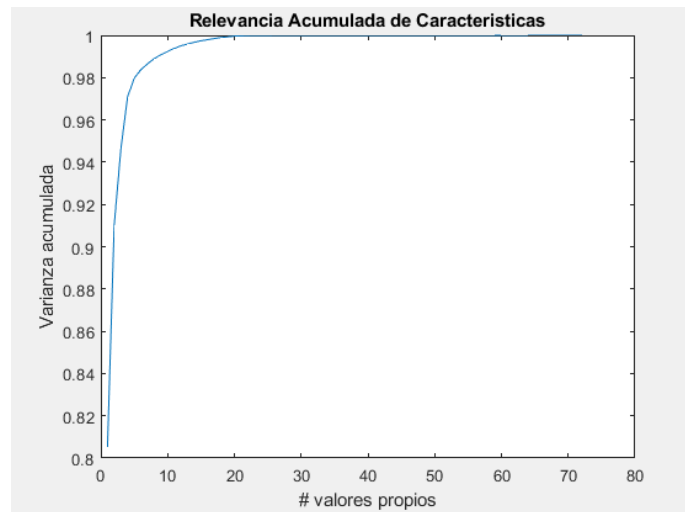


Figura 12. Representación gráfica del análisis de componentes donde las características son representadas en el eje X, la varianza en el eje Y donde se observa que 20 características son las que presentan la mayor varianza.

7.5.3 Correlación

Es una técnica estadística que indica si hay relación o no entre variables o características, es decir que dos características pueden tener alta relación ya sea proporcional o inversamente proporcional o quizá ninguna relación [71] de esta forma indica cuales datos son los que se deben tener en cuenta a través de los coeficientes de correlación como se muestra en la figura 13.

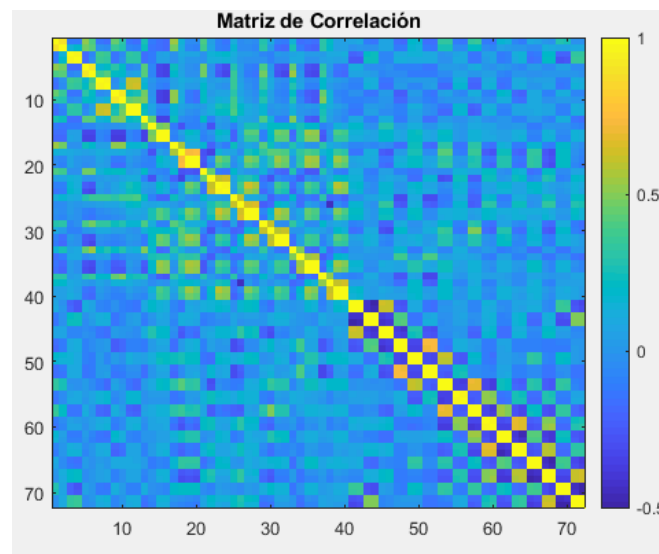


Figura 13. Matriz de correlación entre características donde los coeficientes varían de -0.5 a 1 representado por una escala de color donde si el color es muy amarillo quiere decir que está altamente relacionado.

8. DESARROLLO

Para llevar a cabo el análisis de características tiempo frecuencia y posteriormente la clasificación de 12 señas se tienen en cuenta las pautas mostradas en la figura 14.

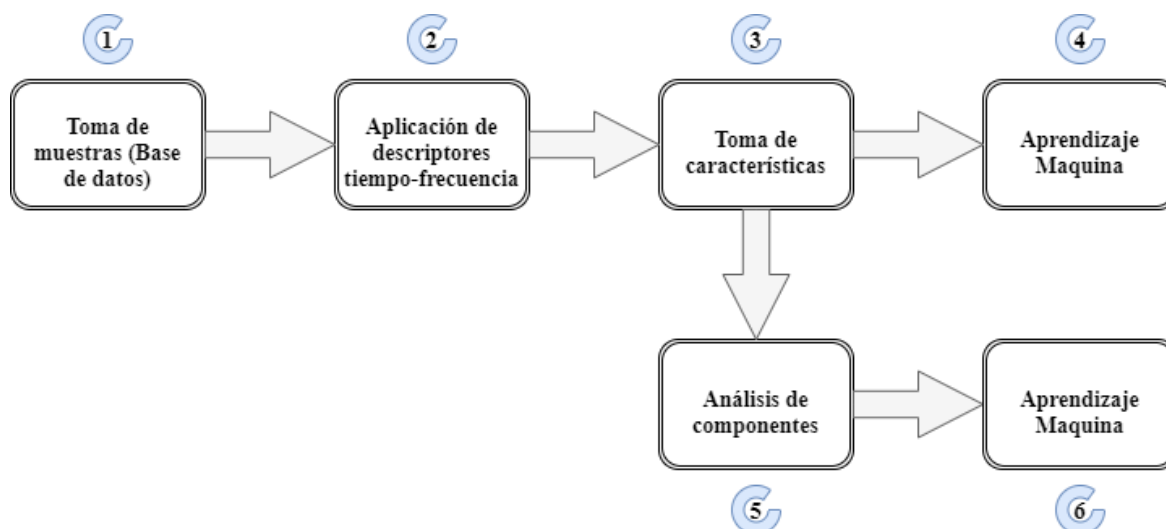


Figura 14. Diagrama de bloques que muestra la metodología a seguir para llevar a cabo la clasificación de señas.

8.1 Base de datos

8.1.1 Señas

Para la toma de los datos se tuvieron en cuenta 12 señas las cuales fueron tomadas del diccionario de lenguaje señas colombiano [2], las señas usadas se muestran en la figura 15.

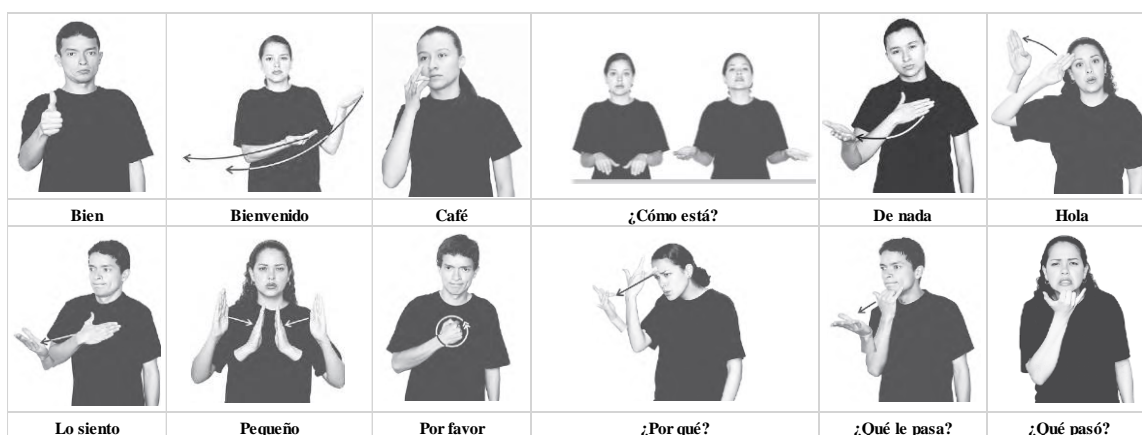


Figura 15. Señas utilizadas para formar la base de datos [2].

8.1.2 Toma de señas

Para esto se debe tener en cuenta que la banda Myo cuenta con una estructura que le permite adaptarse a diferentes tamaños, por lo tanto, puede ser utilizado en diferentes partes del cuerpo. Teniendo en cuenta el desarrollo del presente trabajo el mejor lugar para su utilización es en el antebrazo como se muestra en la figura 16, ya que esta zona es susceptible al movimiento del brazo y a la flexión de los dedos, lo que permite que los sensores capturen la información sin problemas [72].



Figura 16. Posicionamiento para la banda Myo [73].

Para la creación de la base de datos se manejan las 12 señas mostradas en la figura 15 de las cuales se toman 10 muestras a cada una, este proceso se repite para 10 sujetos de prueba de los cuales 6 son hombres y 4 son mujeres; dicha estructura se muestra en la figura 17 dando 1200 muestras tomadas.

Personas		SEÑAS											
		Bien	Bienvenido	Café	¿Como esta?	De nada	Hola	Lo siento	Pequeño	Por favor	¿Por que?	¿Que le pasa?	¿Que paso?
		No. muestras	No. muestras	No. muestras	No. muestras	No. muestras	No. muestras	No. muestras	No. muestras	No. muestras	No. muestras	No. muestras	No. muestras
1		10	10	10	10	10	10	10	10	10	10	10	10
2		10	10	10	10	10	10	10	10	10	10	10	10
3		10	10	10	10	10	10	10	10	10	10	10	10
4		10	10	10	10	10	10	10	10	10	10	10	10
5		10	10	10	10	10	10	10	10	10	10	10	10
6		10	10	10	10	10	10	10	10	10	10	10	10
7		10	10	10	10	10	10	10	10	10	10	10	10
8		10	10	10	10	10	10	10	10	10	10	10	10
9		10	10	10	10	10	10	10	10	10	10	10	10
10		10	10	10	10	10	10	10	10	10	10	10	10

Figura 17. Estructura de la base de datos.

Para la toma de señas se utiliza el entorno de desarrollo *Matlab*[®] y la librería¹² entregada para este entorno que permite la captura de datos provenientes de cada uno de los sensores de la banda Myo. Con esta librería se implementa una interfaz gráfica de usuario mostrada en la figura 18 que permite dar inicio a la toma de cada una de las muestras y brindarles un nombre, además se le define un tiempo aproximado de 3 segundos para la reproducción de la seña y una frecuencia de muestreo de 200Hz para mantener la misma frecuencia que maneja la banda, en la figura 19 se muestra el diagrama de flujo que explica el algoritmo.

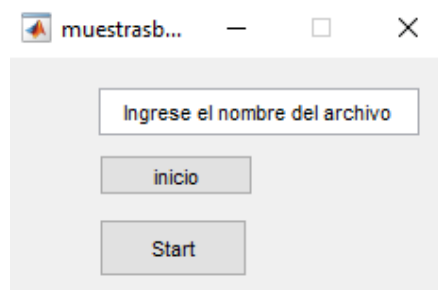


Figura 18. Interfaz gráfica de usuario o GUI donde se puede nombrar las señas tomas, el botón de “inicio” permite reconocer si hay conexión con la banda Myo y el botón “Start” inicia la toma de la muestra.

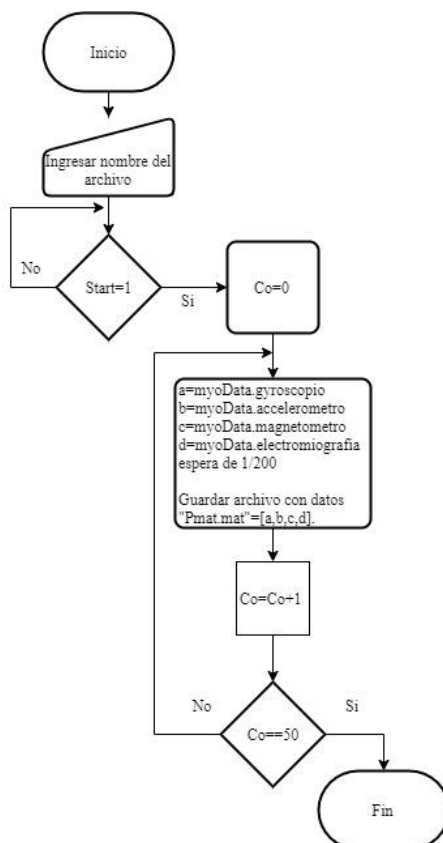


Figura 19. Diagrama de flujo para la toma de datos.

¹² Esta librería fue provista por Mark Tomaszewski y toda la información correspondiente a esta, así como su uso se encuentra de manera libre en la página de MathWorks[®] [74].

8.2 Aplicación de descriptores tiempo-frecuencia

Para la aplicación de los descriptores se debe tener en cuenta que los archivos con extensiones “. mat” guardados como se indica en la figura 19 contienen la información de los diferentes sensores, alojada en variables “a,b,c,d” pero cada uno de los sensores tienen diferentes ejes representados como canales en los vectores obtenidos como se muestra en la figura 20.

Muestra																	
Acelerometro			Giroscopio			Magetometro				Electromiografia							
Canales			Canales			Canales				Canales							
1	2	3	1	2	3	1	2	3	4	1	2	3	4	5	6	7	8
Datos del canal			Datos del canal			Datos del canal				Datos del canal					Datos del canal		
Datos del canal			Datos del canal			Datos del canal				Datos del canal					Datos del canal		
Datos del canal			Datos del canal			Datos del canal				Datos del canal					Datos del canal		

Figura 20. Distribución de datos por sensor en cada muestra.

8.2.1 FFT

Para aplicar la Transformada Rápida de Fourier (FFT) se debe tener en cuenta que la función “FFT ()” entrega valores reales y complejos. Debido a que solo se necesita su magnitud para análisis se aplica el valor absoluto del resultado obtenido. La FFT se aplica a los datos de cada canal (Observar figura 20) por tanto toda la información de cada señal debe ser organizada para facilitar el algoritmo para esto se organiza en estructuras¹³ por señal como se muestra en la figura 21.

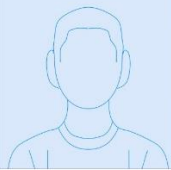

Estructura para señal "Bien"											
Persona 1			Muestra 1								
			
			
			
			
			
			
			
			
			
			
Persona 10			Muestra 1								
			
			
			
			
			
			
			
			
			
			

Figura 21. Estructura para la señal Bien.

¹³ Una estructura se considera como un arreglo de datos el cual puede contener datos de diferentes tipos como números, cadena de caracteres, desimánales, vectores, listas, otras estructuras entre otros.

Luego de organizar la información se ingresa en las estructuras para las 12 señas y a su vez se aplica la FFT la cual se encuentra en el entorno de desarrollo *Matlab*[®] como “fft”, a cada canal de las 100 muestras por estructura, en la figura 22 se observa el diagrama de flujo del algoritmo.

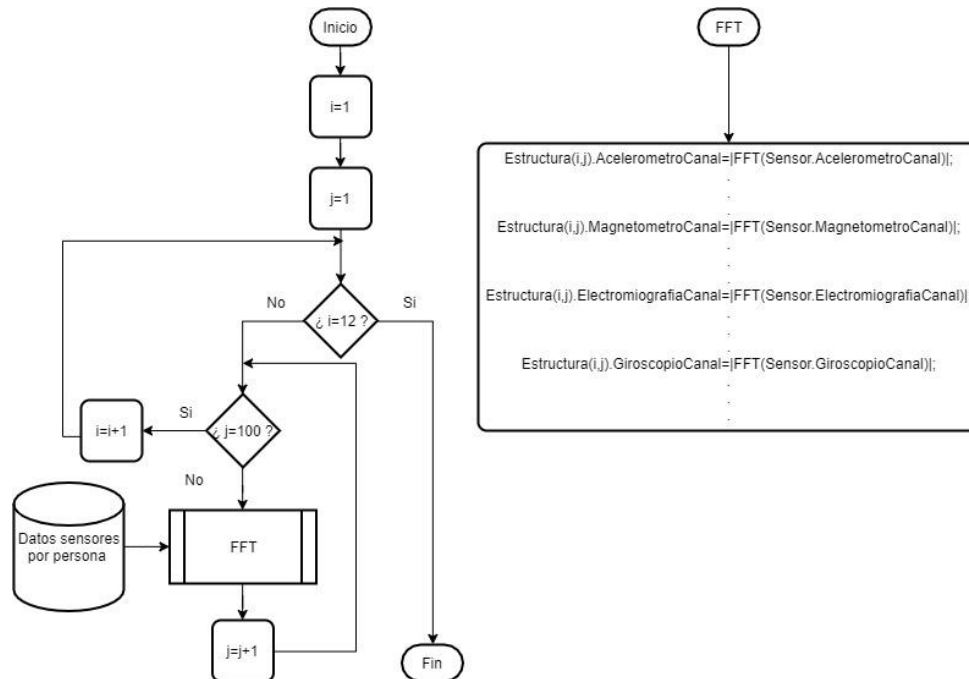


Figura 22. Diagrama de flujo para aplicar la FFT donde el contador “i” indica el cambio entre estructuras y “j” el cambio entre muestras.

Con el fin de organizar los datos para los clasificadores los datos obtenidos al aplicar la FFT son organizados en una nueva estructura en el orden que se muestra en la figura 23.

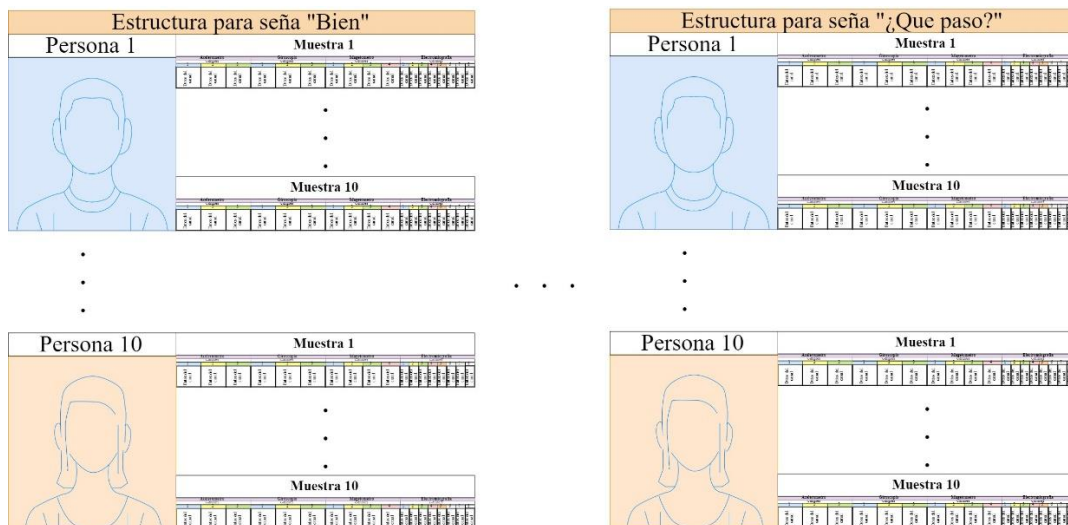


Figura 23. Estructura para las 12 señas comenzando desde la seña “Bien” hasta la seña “¿Qué paso?”.

8.2.2 Wavelet

De manera similar que en la FFT los datos son organizados como en la figura 21 para aplicar la Transformada Wavelet cuya función en el entorno de desarrollo *Matlab*[®] esta dada como “dwt”, por otro lado, la aplicación de esta tiene como resultante dos patrones por tanto la estructura resultante será de igual tamaño, pero contendrá más datos, en la figura 24 se muestra el diagrama de flujo para el algoritmo.

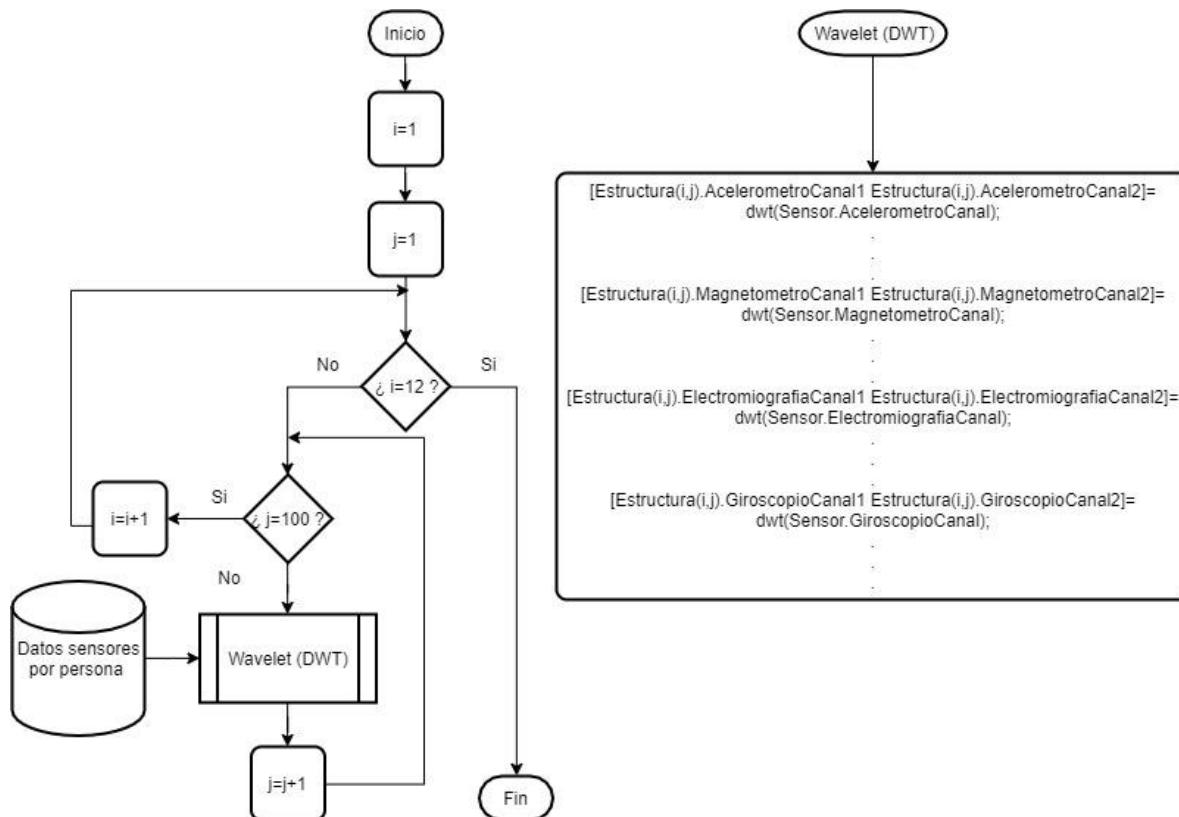


Figura 24. Diagrama de flujo para aplicar DWT donde el contador “i” indica el cambio entre estructuras y “j” el cambio entre muestras.

8.2.3 STFT

Para el caso de la Transformada corta de Fourier o “ventaneada” al igual que para los anteriores descriptores en frecuencia se organizan los datos de manera estructural y se utiliza la función provista por el entorno de desarrollo *Matlab*[®] “stft()” por otro lado se utiliza una de las ventanas que el entorno sugiere (Kaiser [75]), dando como resultado una estructura como la vista en la figura 23 con más información por canal debido a la configuración de la misma, en la figura 25 se observa el diagrama de flujo para el algoritmo.

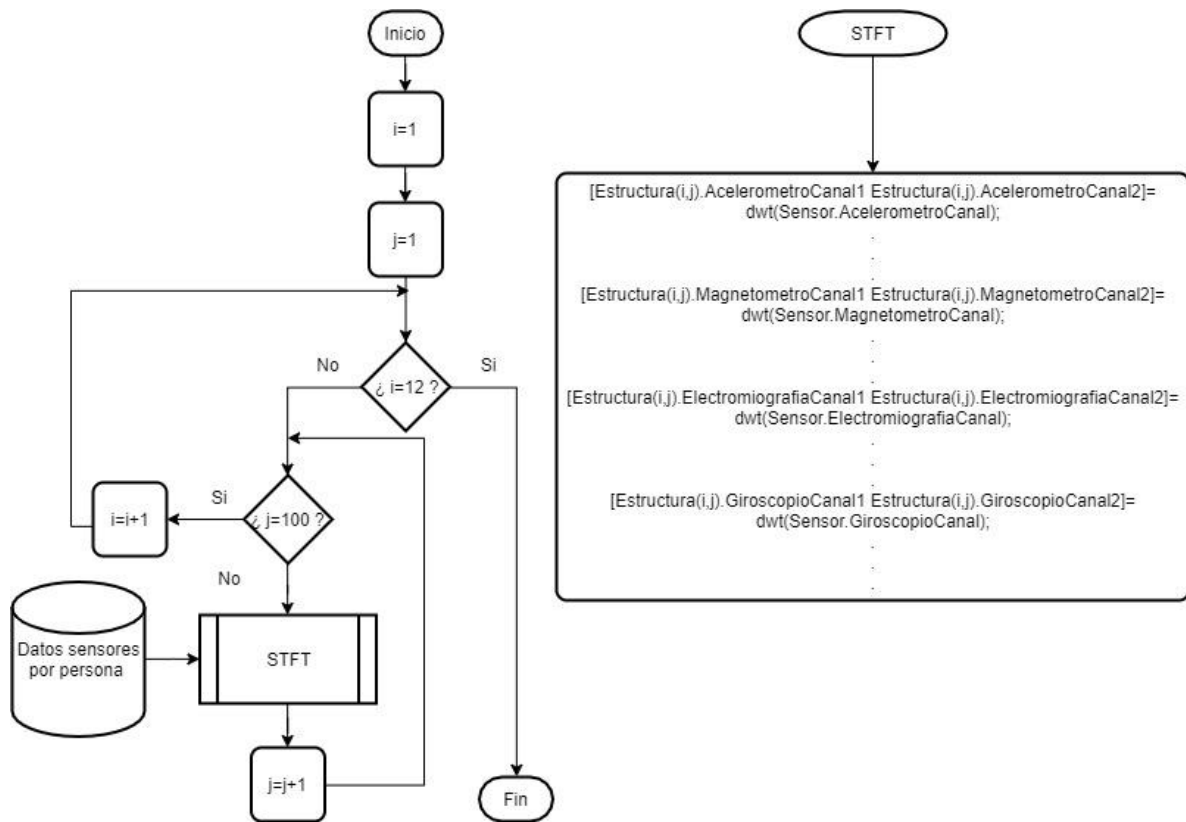


Figura 25. Diagrama de flujo para la aplicación de la STFT donde el contador “i” indica el cambio entre estructuras y “j” el cambio entre muestras.

8.3 Toma de características.

Las características que se tomaron en cuenta fueron la media, la desviación estándar, curtosis y sesgo, las cuales fueron sacadas a cada canal de las estructuras resultantes de aplicar los descriptores tiempo-frecuencia como se muestra en la figura 26 y el diagrama de flujo para el algoritmo se muestra en la figura 27. Las funciones correspondientes en el entorno de desarrollo *Matlab*[®] son “mean (media), std (desviación estandar), kurtosis (curtosis), skewness (sesgo)”.

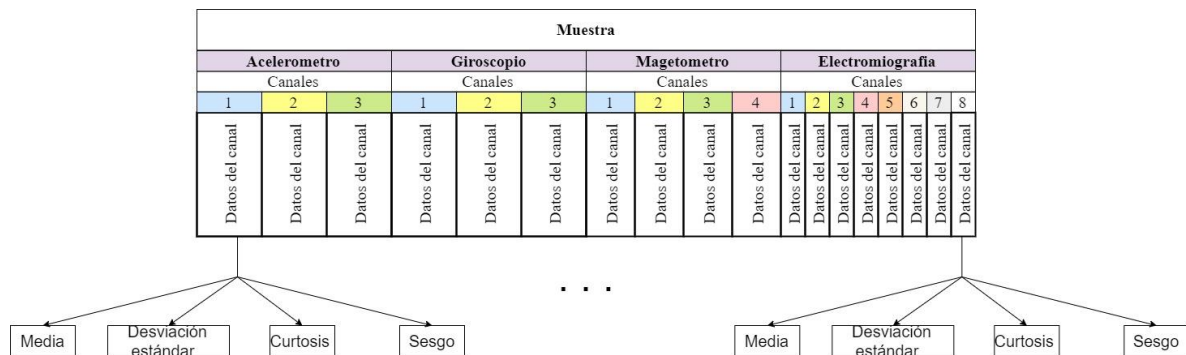


Figura 26. Toma de características por canal

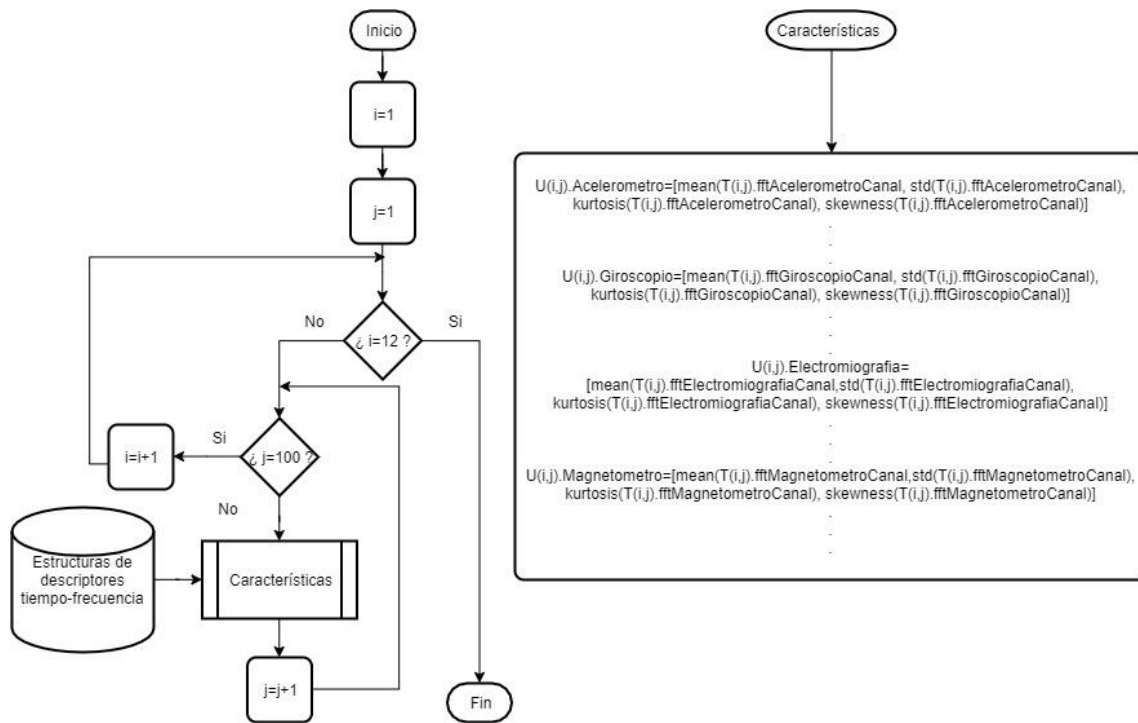


Figura 27. Diagrama de flujo para obtener las características de las estructuras de los descriptores tiempo-frecuencia

8.4 Aprendizaje máquina

Para la aplicación de máquinas de soporte vectorial (SVM) y redes neuronales es necesario organizar en arreglos los datos alojados en las estructuras, teniendo en cuenta que los arreglos solo contendrán la información de un solo tipo de sensor como se observa en la figura 28, esto con el fin de utilizar dichos arreglos para analizar la información sensor por sensor o incluso diferentes combinaciones entre los mismos, además se aplicara a cada descriptor tiempo-frecuencia de manera independiente.

Persona 1	Muestra1											
	Acelerometro											
	Canales											
	1				2				3			
	Media	Desviación estándar	Curtosis	Sesgo	Media	Desviación estándar	Curtosis	Sesgo	Media	Desviación estándar	Curtosis	Sesgo
Persona 10	Muestra100											
	Acelerometro											
	Canales											
	1				2				3			
	Media	Desviación estándar	Curtosis	Sesgo	Media	Desviación estándar	Curtosis	Sesgo	Media	Desviación estándar	Curtosis	Sesgo

Figura 28. Estructura de los arreglos para el aprendizaje máquina para una señal.

8.4.1 SVM

Para la aplicación de las SVM se utiliza una APP (aplicación) con la que cuenta el entorno de desarrollo *Matlab*[®] llamada *Classification Learner*, para el cual solo es necesario el ingreso de los arreglos compuestos por las características y todas las señales al igual que una

etiqueta al final del arreglo que permita identificar a que seña pertenece, en la figura 29 se muestra cómo utilizar la app.

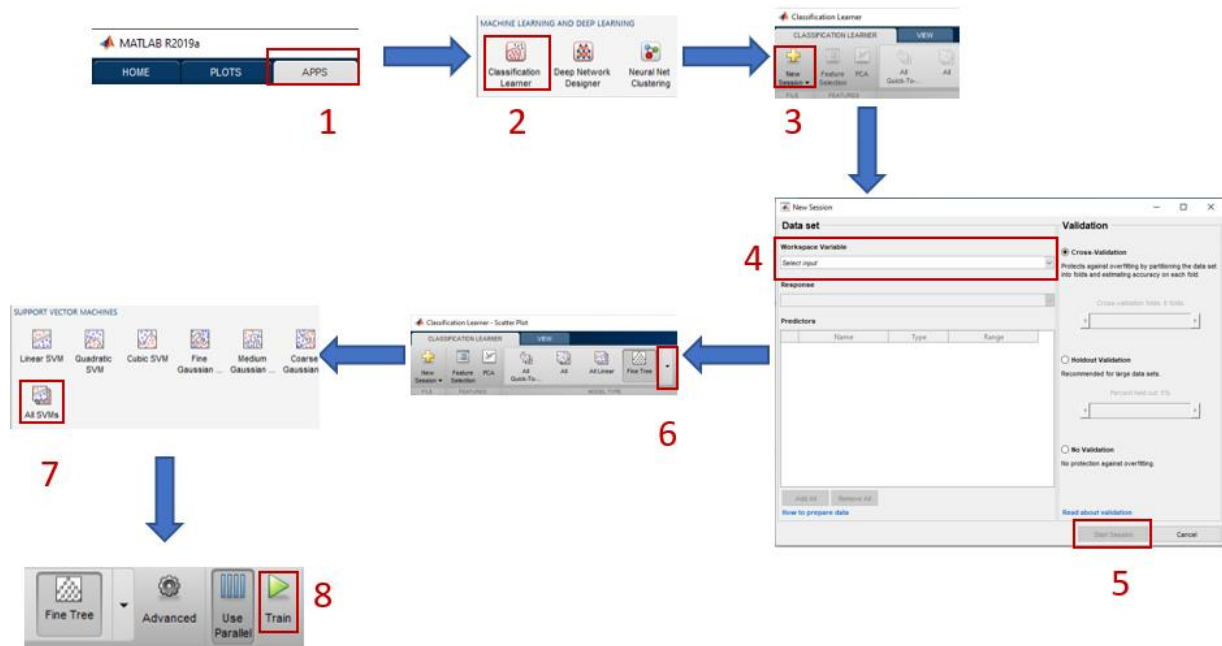


Figura 29. Pasos a seguir para el entrenamiento con SVM donde: 1. Pestaña de APPS, 2. Classification Learner (Aprendizaje maquina), 3. New Session, 4. Workspace Variable (selección arreglo de características), 5. Start Session, 6. Selección de metodos de Machine Learning, 7. ALL SVMs (aplica diferentes kernel), 8. Train (Inicio de entrenamiento).

8.4.2 Redes neuronales

Para la aplicación de redes neuronales al igual que las SVM se utiliza una APP provista por el entorno de desarrollo *Matlab*[®] llamada *Neuronal Net Pattern Recognition*, para la cual es necesario el ingreso de un arreglo de características y un arreglo *Matlab*[®] de etiquetas que permita identificar que seña es la ingresada, pero a diferencia de las SVM dicho arreglo debe ser binario como se muestra en la figura 30 y en la figura 31 se muestra cómo utilizar la app.

1200 filas	1	0	0	0	0	0	0	0	0	0	0	0
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	0	1	0	0	0	0	0	0	0	0	0	0
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	0	0	1	0	0	0	0	0	0	0	0	0
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	0	0	0	0	0	0	0	0	0	0	0	1
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
12 columnas												

Figura 30. Arreglo binario pedido por la app de redes neuronales donde las 12 columnas indican las 12 señas y las 1200 filas indican la cantidad de muestras tomadas a todos los sujetos.

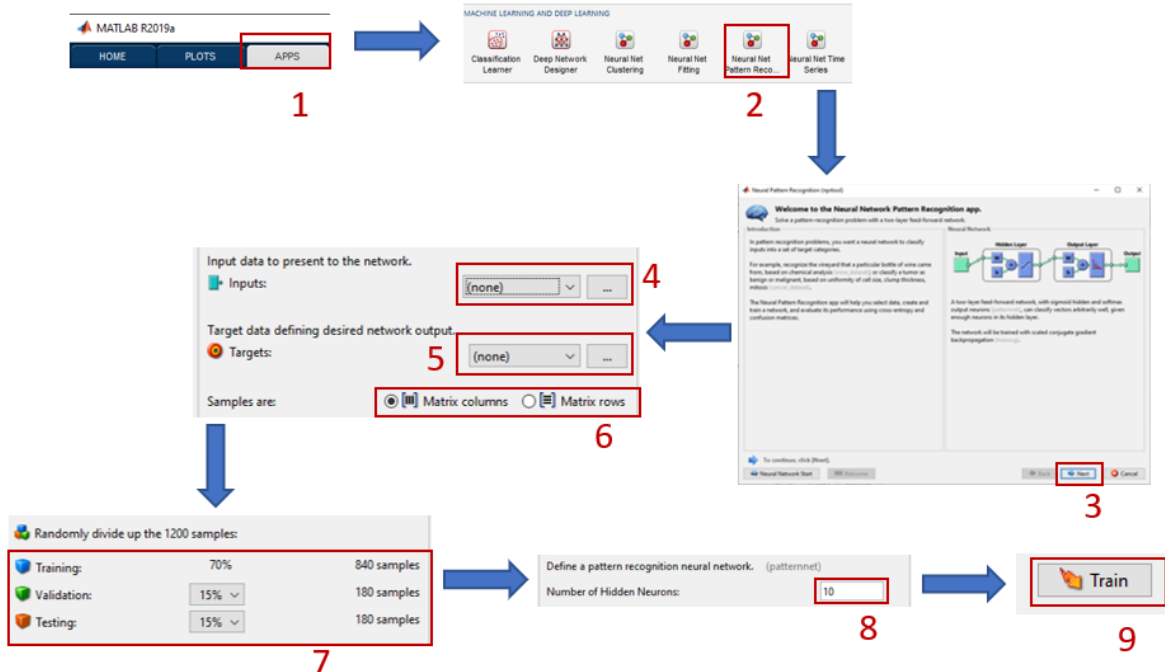


Figura 31. Pasos a seguir para el entrenamiento con Redes Neuronales 31 donde: 1. Pestaña de APPS, 2. Neuronal Net Pattern Recognition (Aprendizaje maquina), 3. Next, 4. Inputs (arreglo de características), 5. Targets (Arreglo binario), 6. Samples are (Especificar si las características son filas o columnas), 7. Training (Porcentaje del arreglo para entrenamiento), Validation (Porcentaje del arreglo para validacion), Testing (Porcentaje del arreglo para testeo), 8. Number of Hidden Neurons (número de capas), 9. Train (Entrenamiento del algoritmo).

8.5 Análisis de componentes

Para el análisis de componentes es necesario la utilización de arreglos como los vistos en la figura 28 que contenga todas las señales para el mismo sensor y algunas funciones como PCA, Correlación y Relieff, para las cuales se utilizan funciones predefinidas provistas por el entorno de desarrollo *Matlab*[®] y la manera de cómo se desarrolla el algoritmo para aplicar las mismas se muestra a continuación.

8.5.1 PCA (Principal Component Analysis)

Para la aplicación de PCA se utiliza como referencia una función programada¹⁴ que permite visualizar la relevancia de los componentes, para esto solo es necesario entregar el arreglo de características perteneciente a todos los sensores, señales y personas, en la figura 32 se puede observar el diagrama de flujo que describe el algoritmo.

¹⁴ La función utilizada para el cálculo del PCA fue provista por el profesor Cristian Andrés Escudero Zapata perteneciente a la Universidad Tecnológica de Pereira.

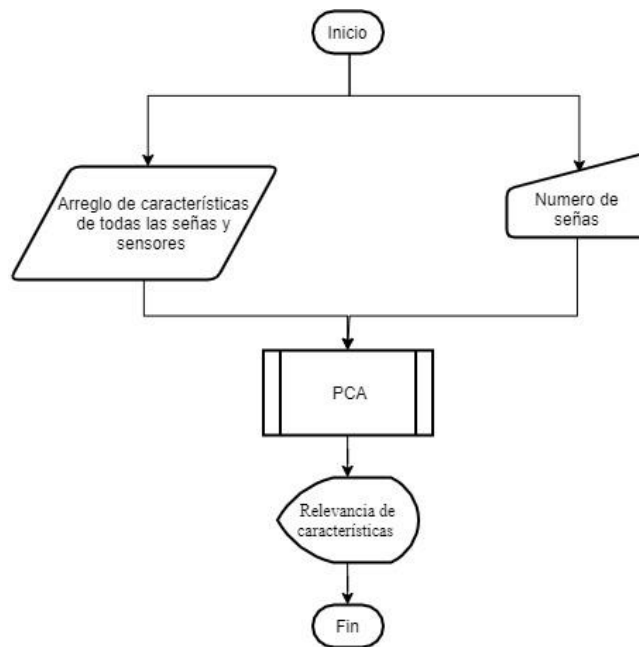


Figura 32. Diagrama de flujo para la utilización de la función PCA.

8.5.2 Correlación

Para aplicar la correlación se utiliza una función provista por el entorno de desarrollo *Matlab*[®] nombrada como “corr” a la cual solo se le entrega el arreglo de características compuesto por todos los sensores, señas y personas, en la figura 33 se muestra el diagrama de flujo que describe el algoritmo.



Figura 33. Diagrama de flujo para la aplicación de la Correlación a el arreglo de características.

8.5.3 Relieff

Para aplicar Relieff se utiliza una función provista por el entorno de desarrollo *Matlab*[®] nombrada como “relieff” a la cual se le entrega el arreglo de características compuesto por todos los sensores, señas y personas, además de, un arreglo con las etiquetas que señalan a que seña pertenecen las características y la cantidad de señas que se utilizaran, en la figura 34 se muestra el diagrama de flujo que describe el algoritmo.

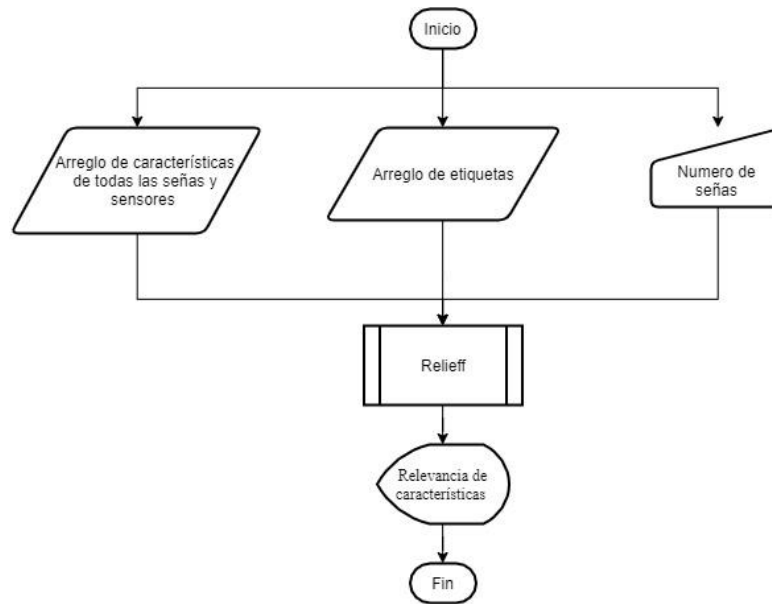


Figura 34. Diagrama de flujo para la aplicación de Relieff a el arreglo de características.

9. ANÁLISIS Y RESULTADOS

A continuación, se presentan los resultados obtenidos aplicando SVM y redes neuronales antes y después del análisis de componentes para el cual se le entregan diferentes arreglos con combinaciones de sensores como se muestran en la figura 35 y las combinaciones ingresadas se muestran en la tabla 1.

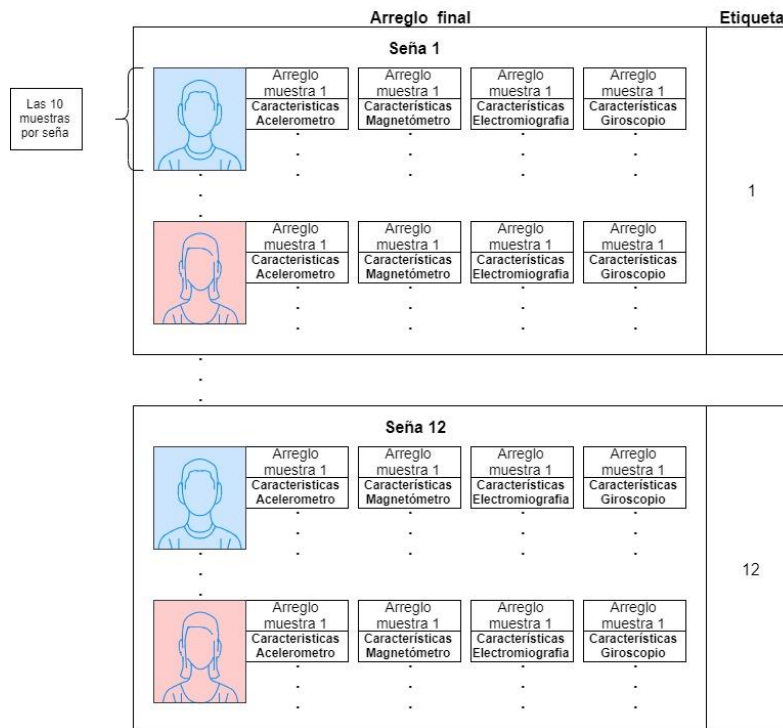


Figura 35. Arreglo ingresado con todos los sensores para en el entrenamiento con SVM.

No.	Combinaciones			
1	Acelerómetro			
2	Giroscopio			
3	Magnetómetro			
4	Electromiografía			
5	Acelerómetro	Giroscopio		
6	Acelerómetro	Magnetómetro		
7	Acelerómetro	Electromiografía		
8	Giroscopio	Magnetómetro		
9	Giroscopio	Electromiografía		
10	Magnetómetro	Electromiografía		
11	Giroscopio	Magnetómetro	Electromiografía	
12	Acelerómetro	Magnetómetro	Electromiografía	
13	Acelerómetro	Giroscopio	Electromiografía	
14	Acelerómetro	Giroscopio	Magnetómetro	
15	Acelerómetro	Giroscopio	Magnetómetro	Electromiografía

Tabla 1. Combinaciones de sensores para entrenamiento

9.1 Resultados obtenidos para SVM

A continuación, se muestran cuáles de los kernel utilizados por la app son los que mejor funcionan para cada uno de los descriptores tiempo-frecuencia usados.

9.1.1 SVM utilizando FFT¹⁵

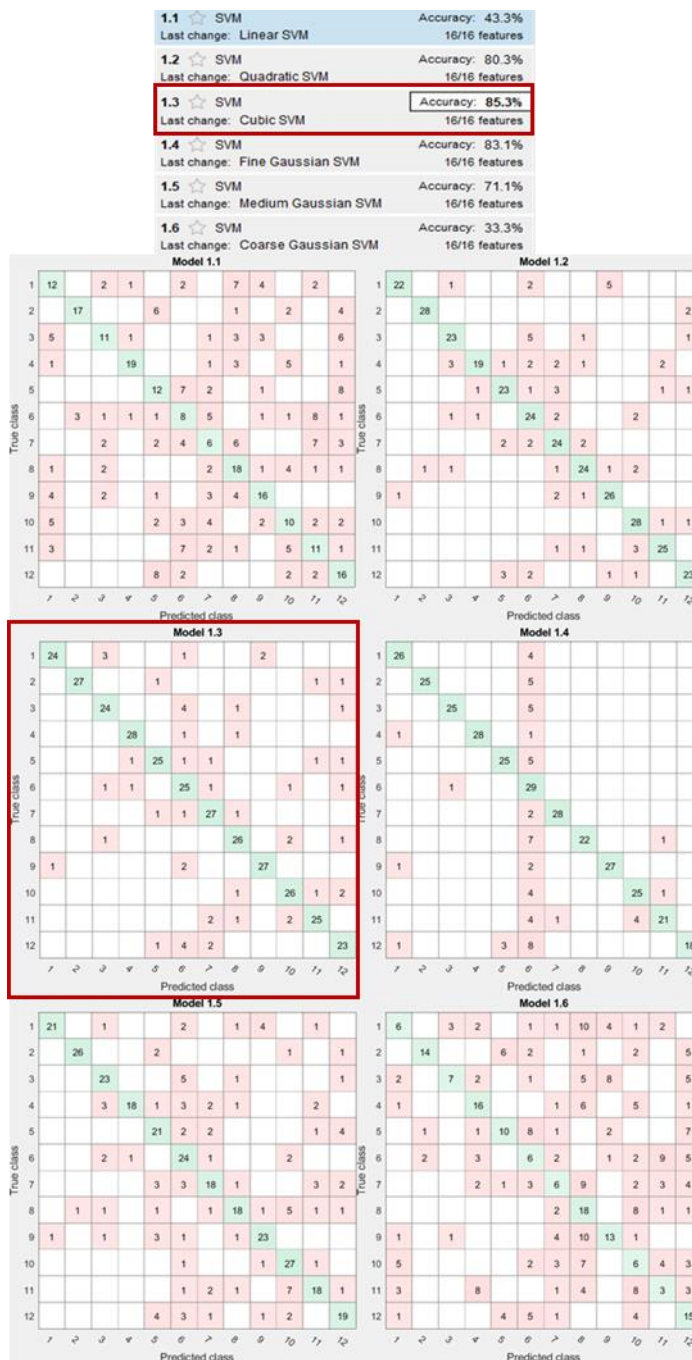


Figura 36. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 3 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

¹⁵ Solo se muestran las combinaciones con los mejores resultados obtenidos.

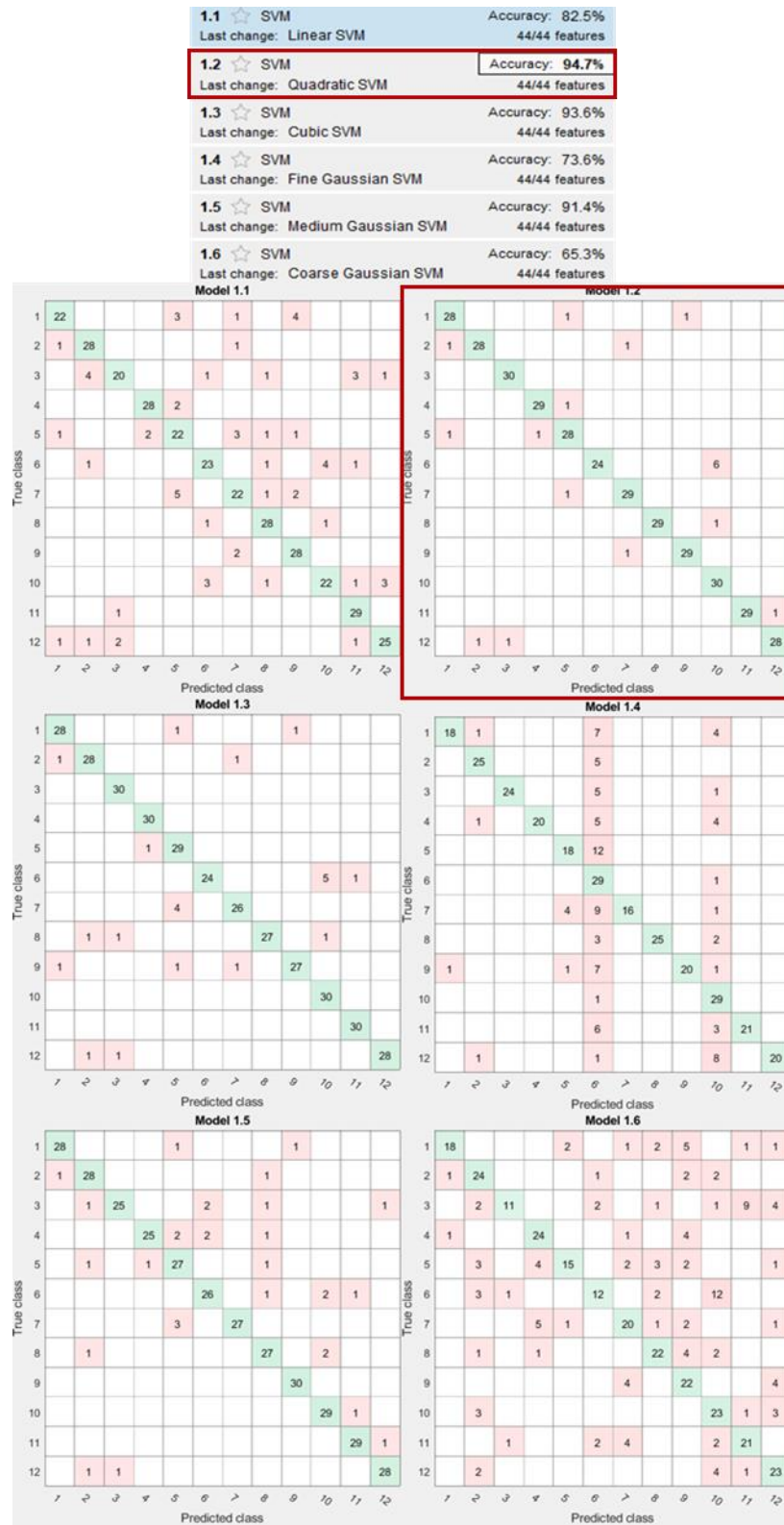


Figura 37. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 9 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2), entrega una mayor numero de aciertos.

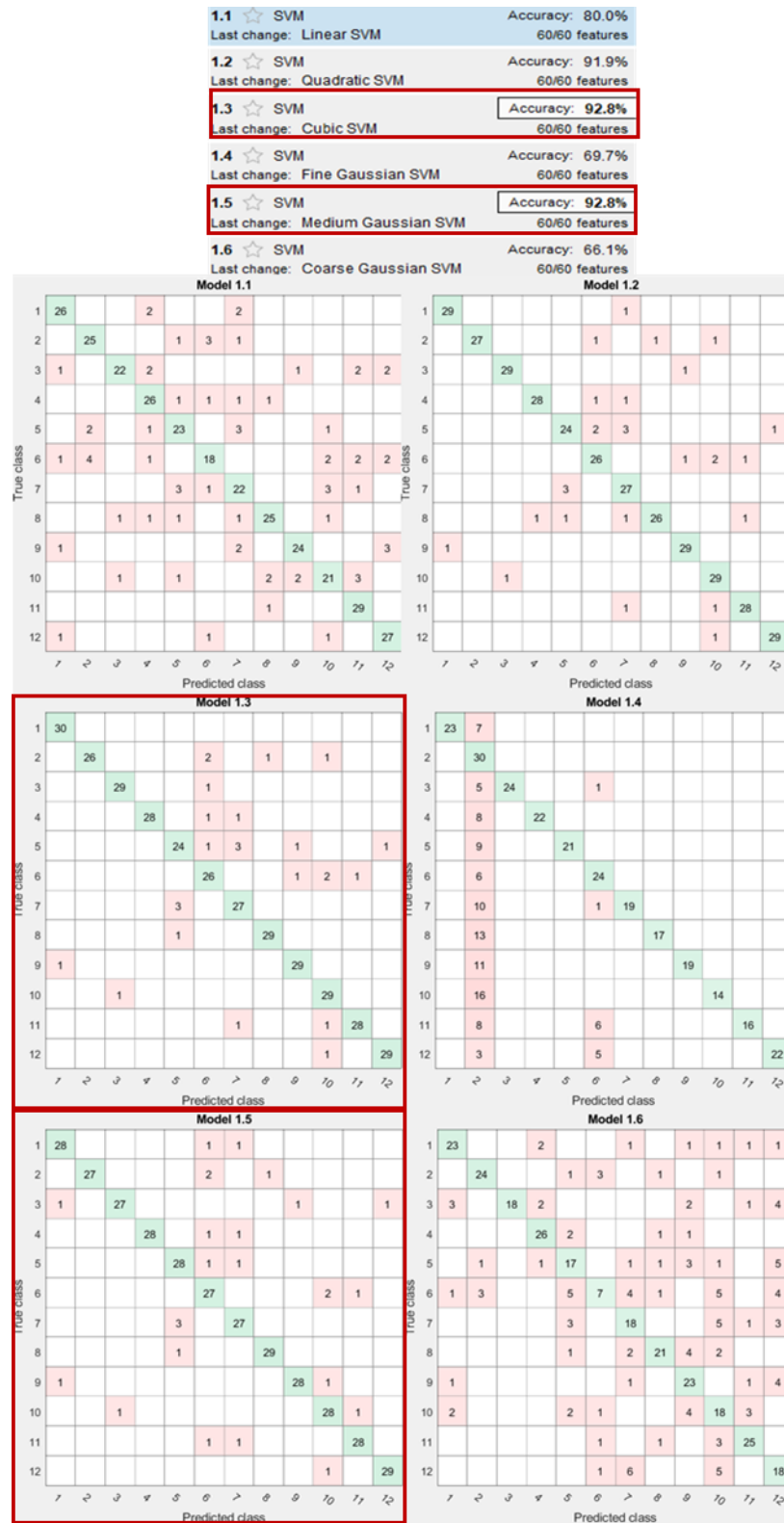


Figura 38. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 12 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3) y el quinto (1.5 Medium Gaussian SVM, Model 1.5), entregan una mayor numero de aciertos.

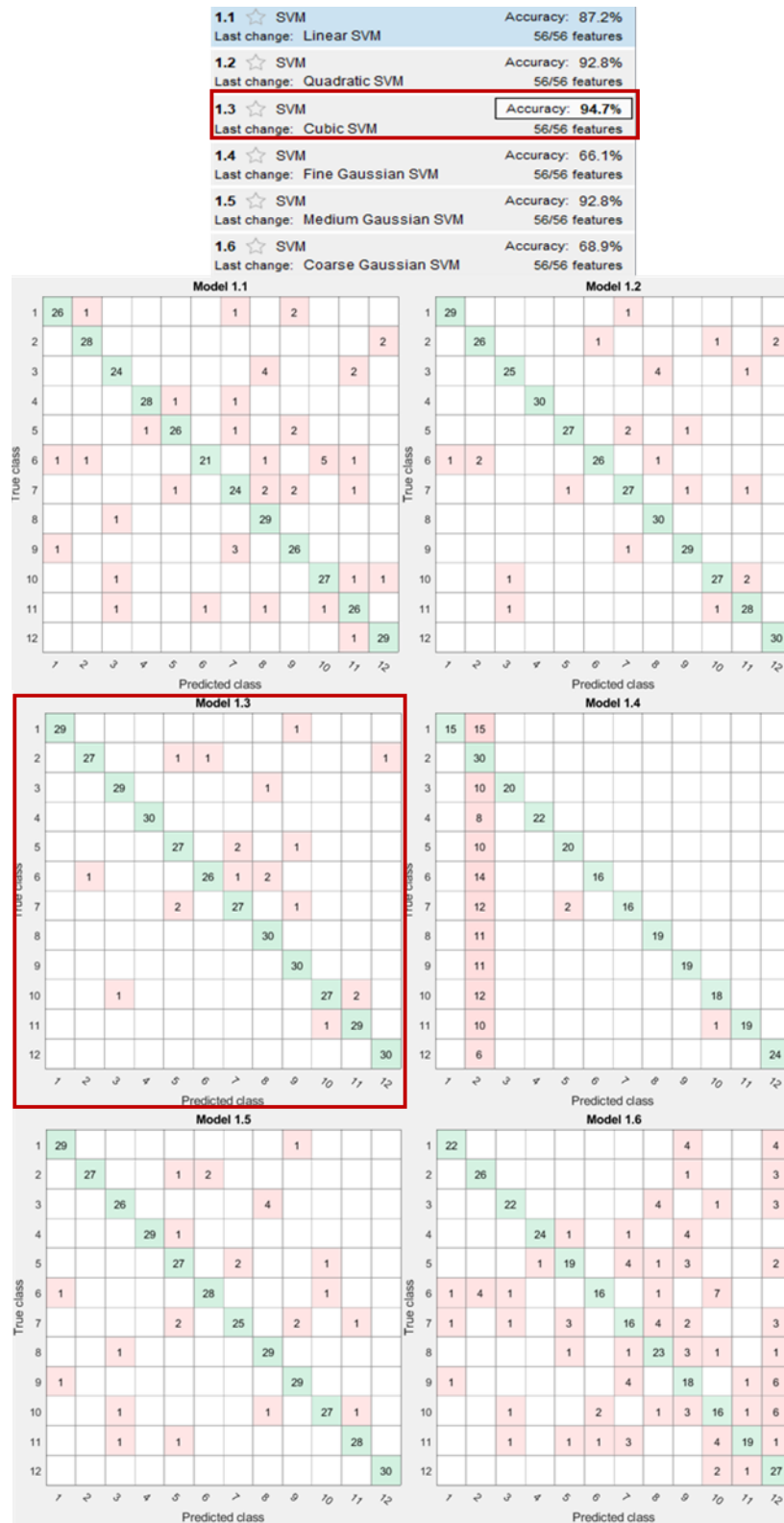


Figura 39. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 13 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

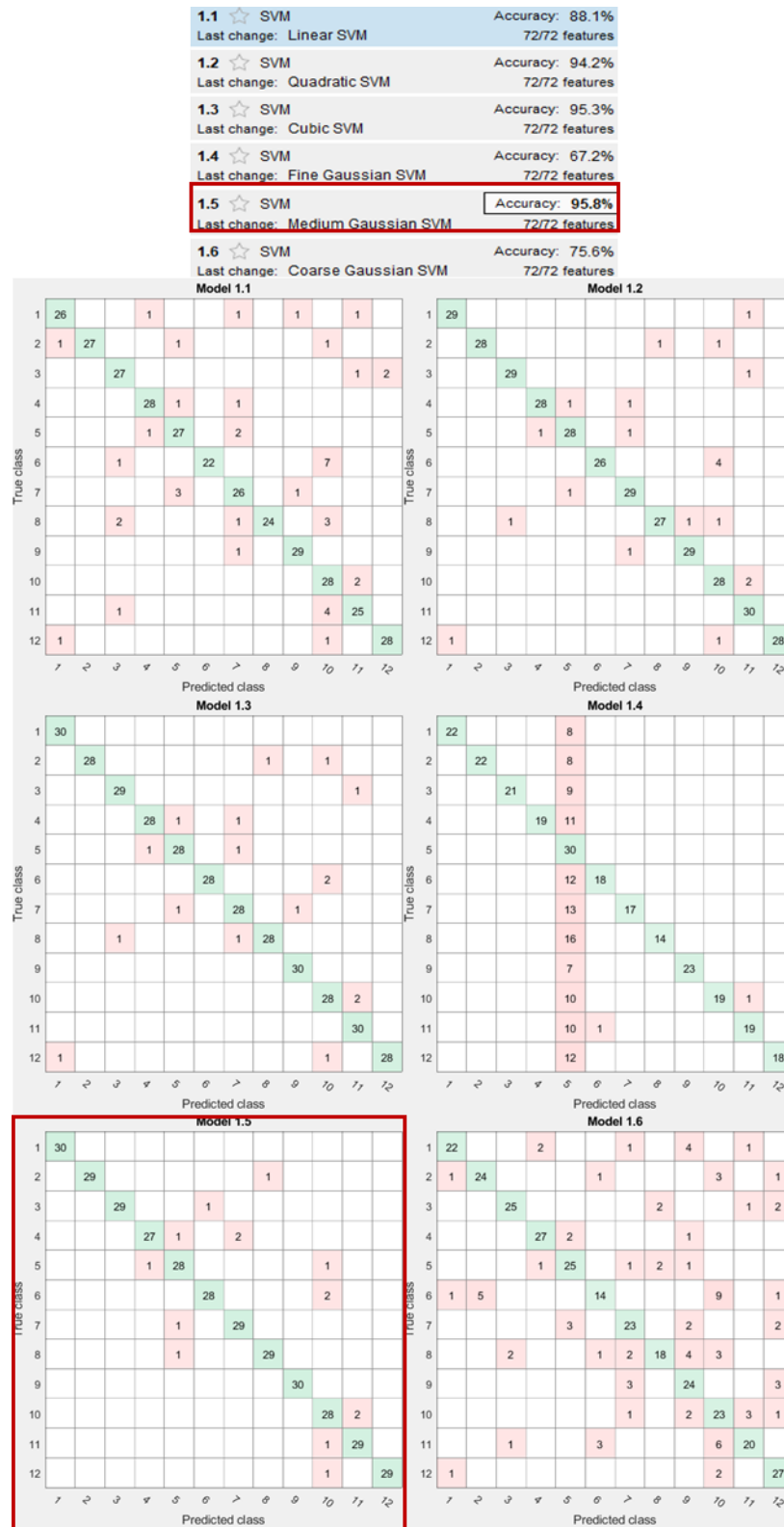


Figura 40. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 15 donde el quinto kernel aplicado (1.5 Medium Gaussian SVM, Model 1.5), entrega una mayor numero de aciertos.

9.1.2 SVM utilizando Wavelet¹⁶

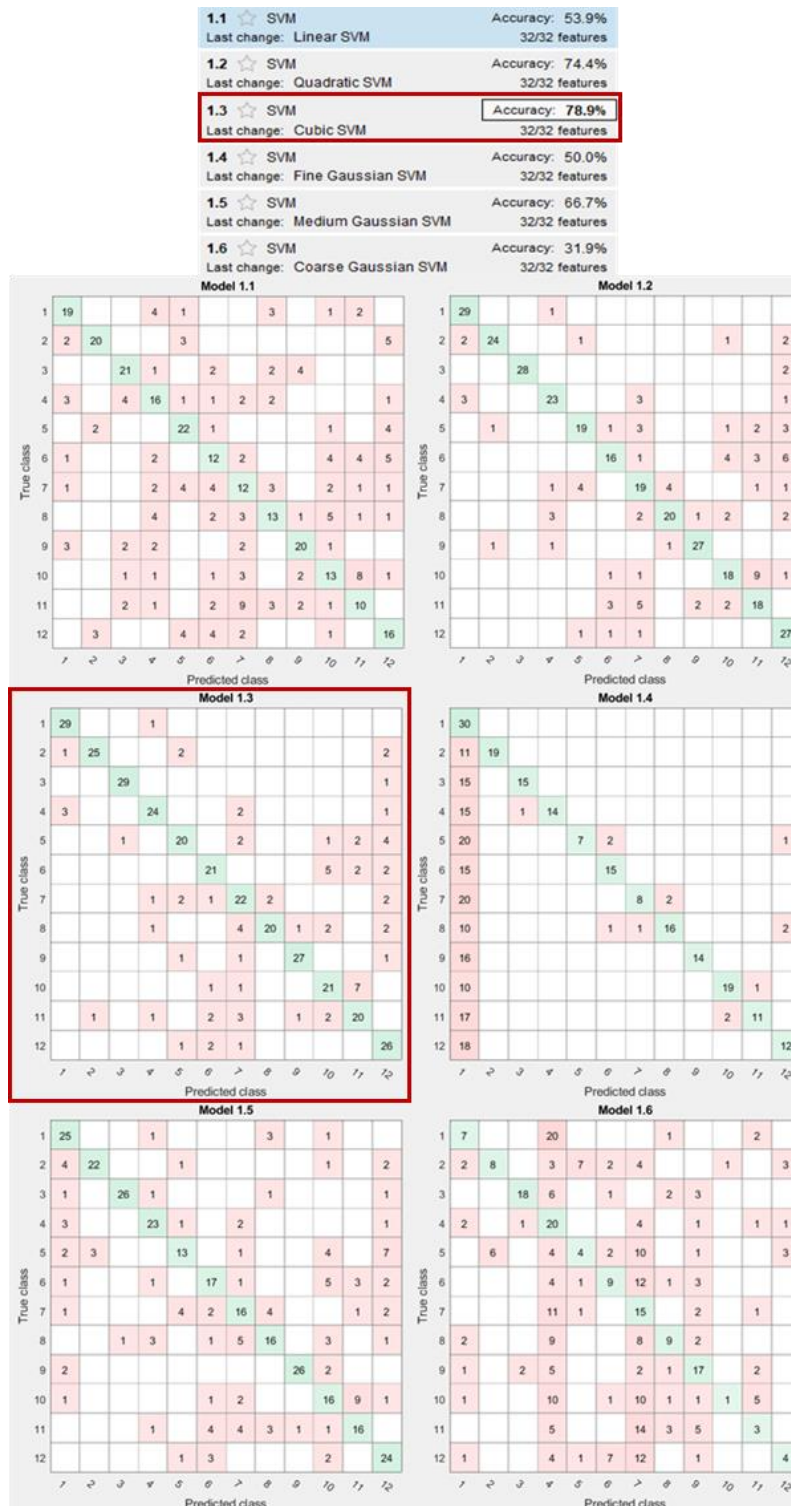


Figura 41. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 3 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

¹⁶ Solo se muestran las combinaciones con los mejores resultados obtenidos.

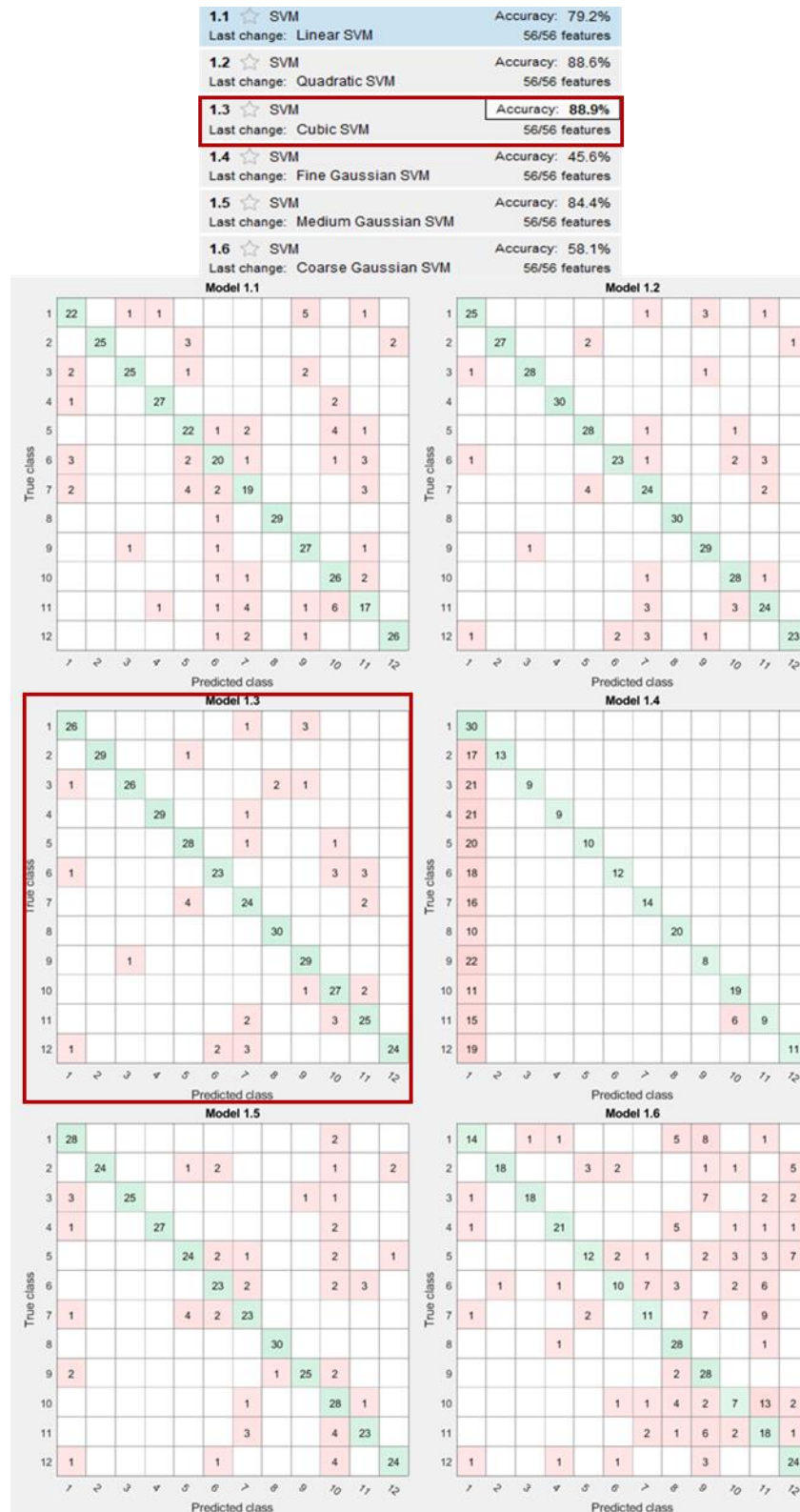


Figura 42. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 6 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

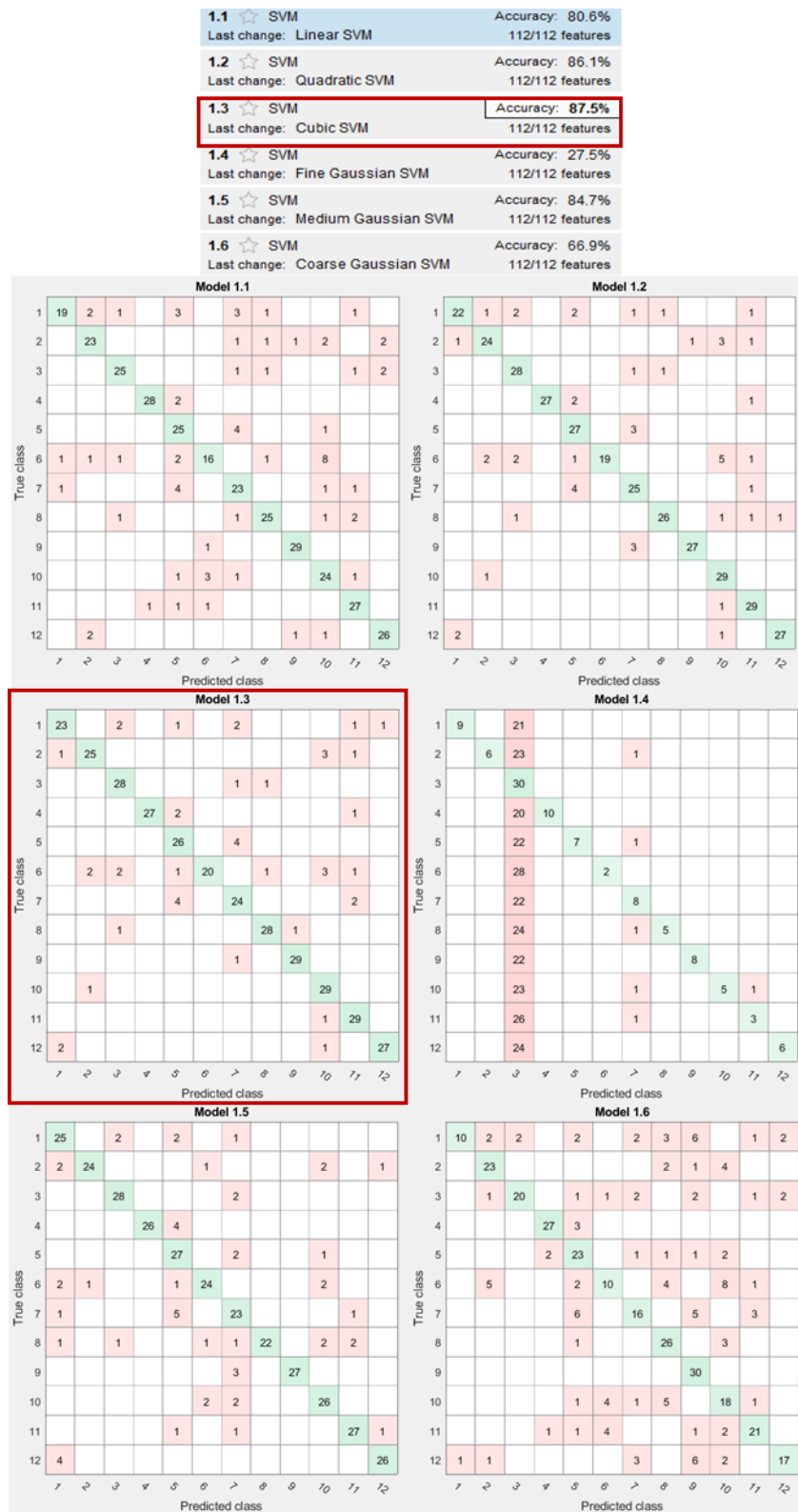


Figura 43. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 13 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

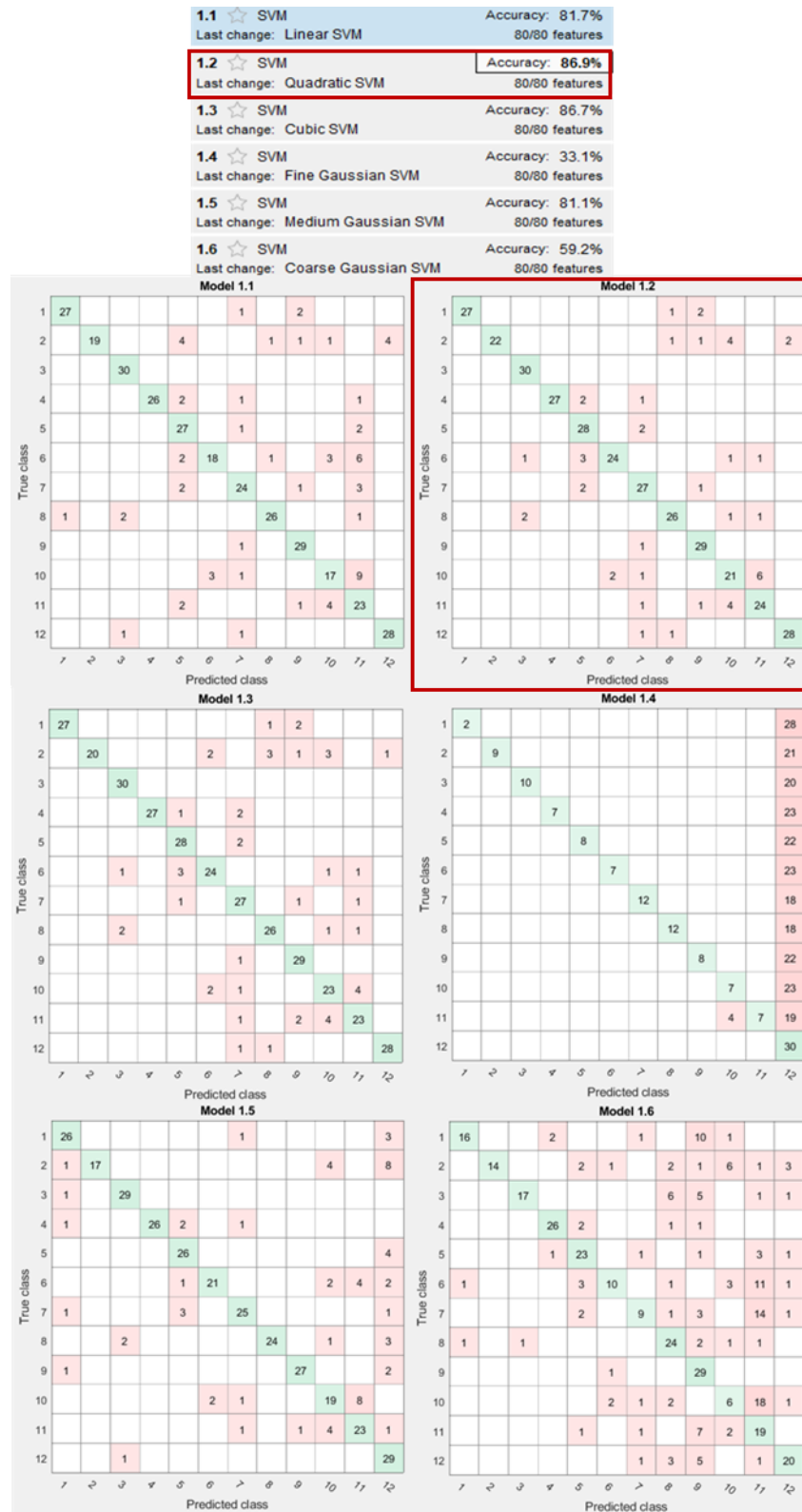


Figura 44. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 14 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2), entrega una mayor numero de aciertos.

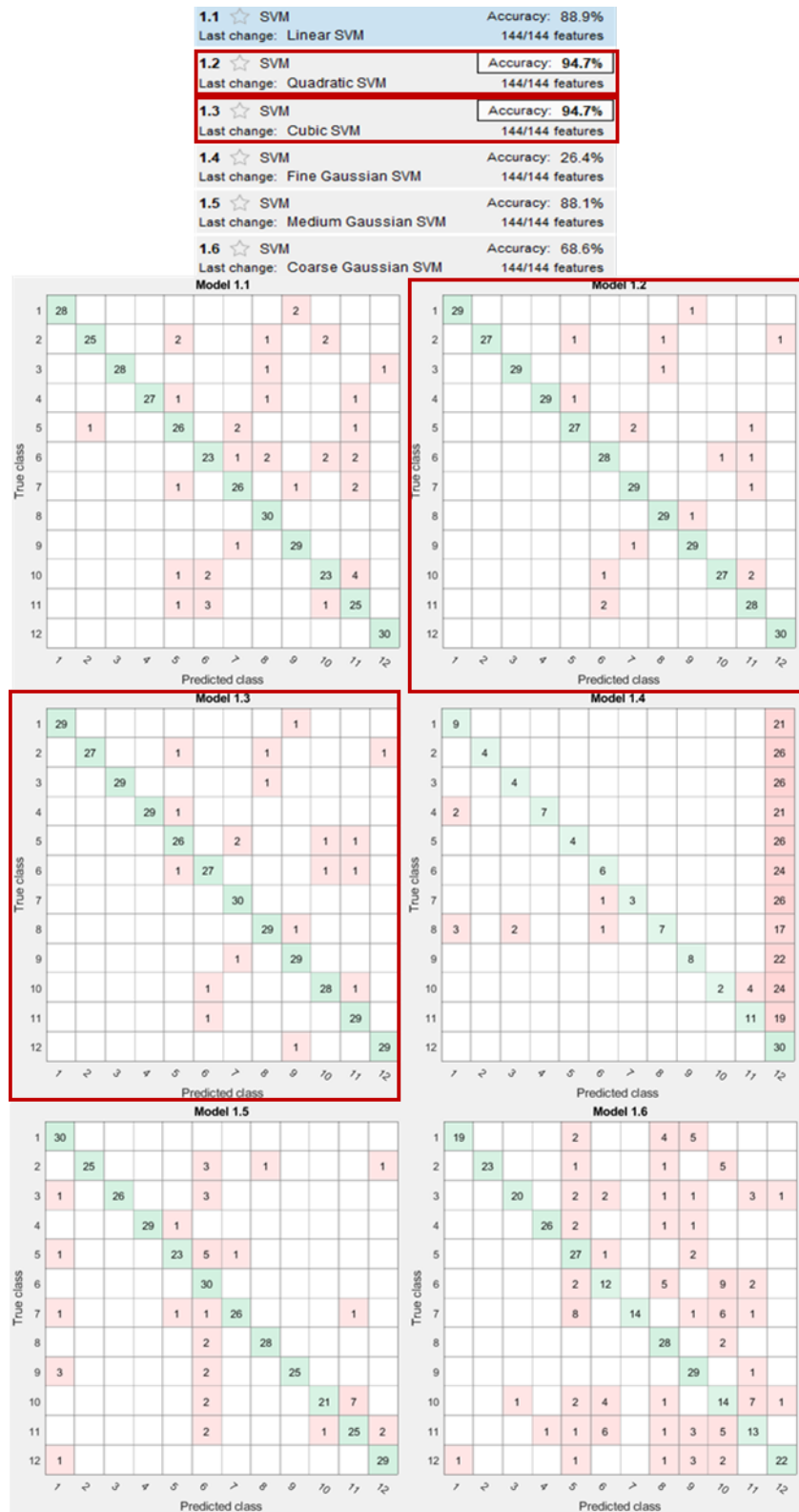


Figura 45. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 15 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2) y el tercero (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

9.1.2 SVM utilizando STFT¹⁷

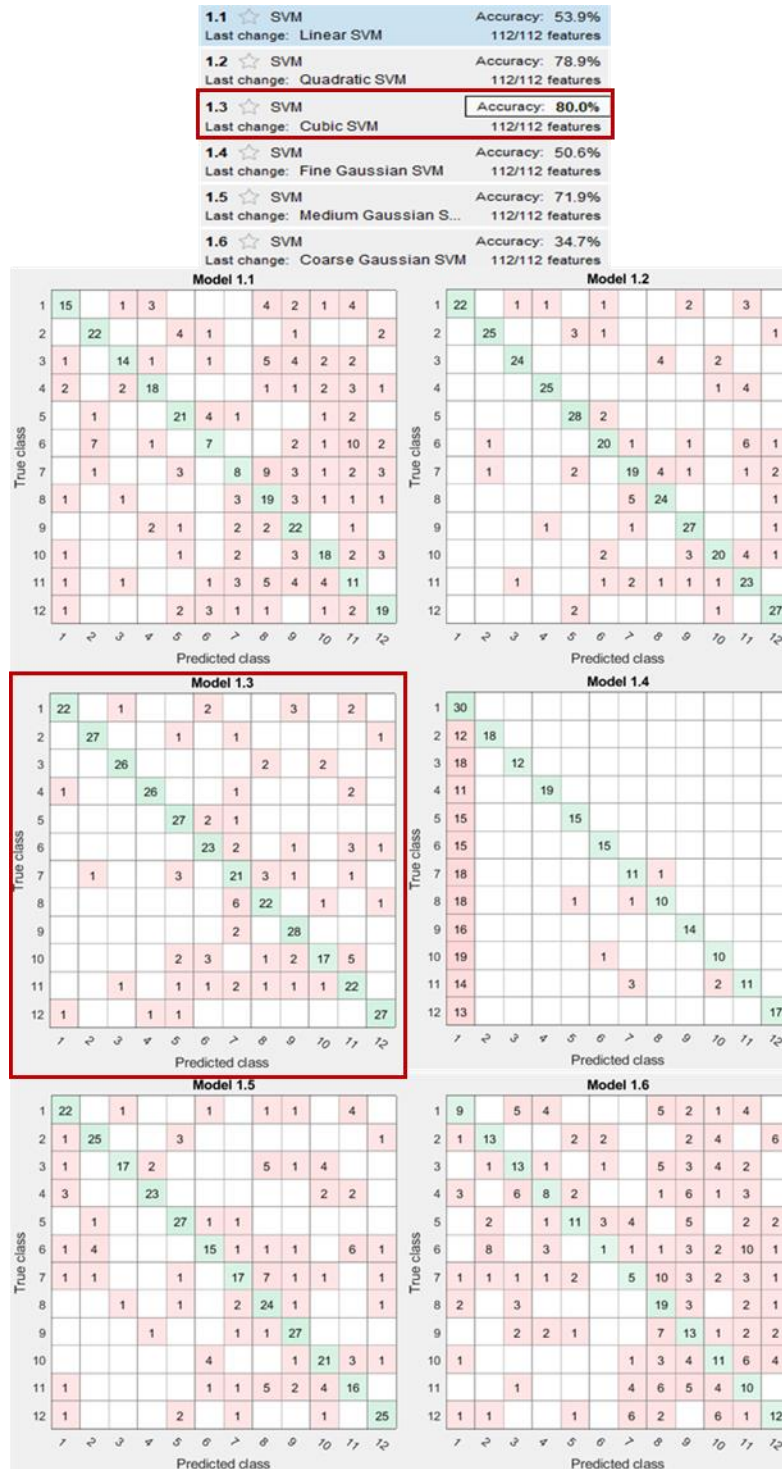


Figura 46. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 3 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

¹⁷ Solo se muestran las combinaciones con los mejores resultados obtenidos.

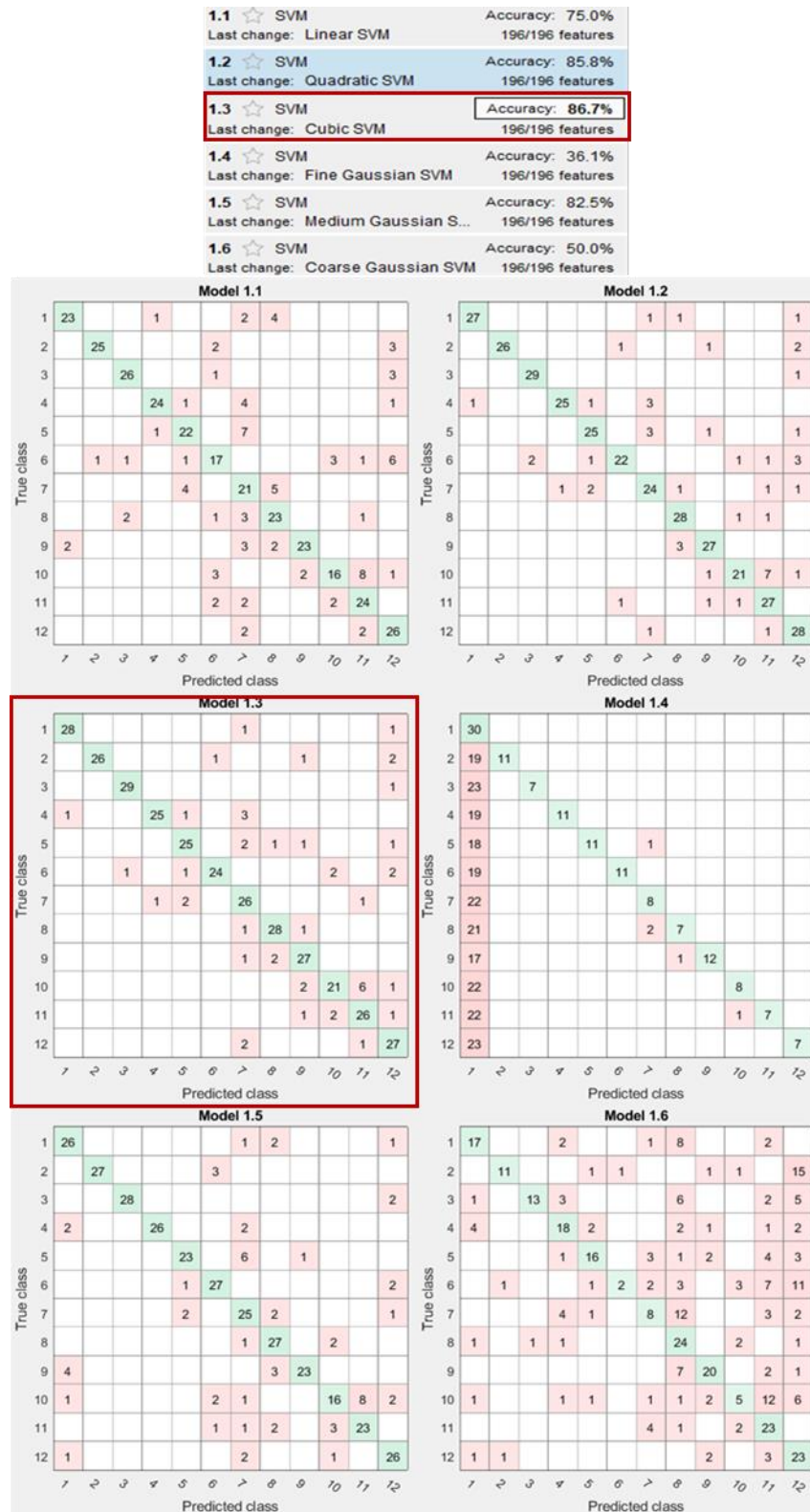


Figura 47. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 8 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2), entrega una mayor numero de aciertos.

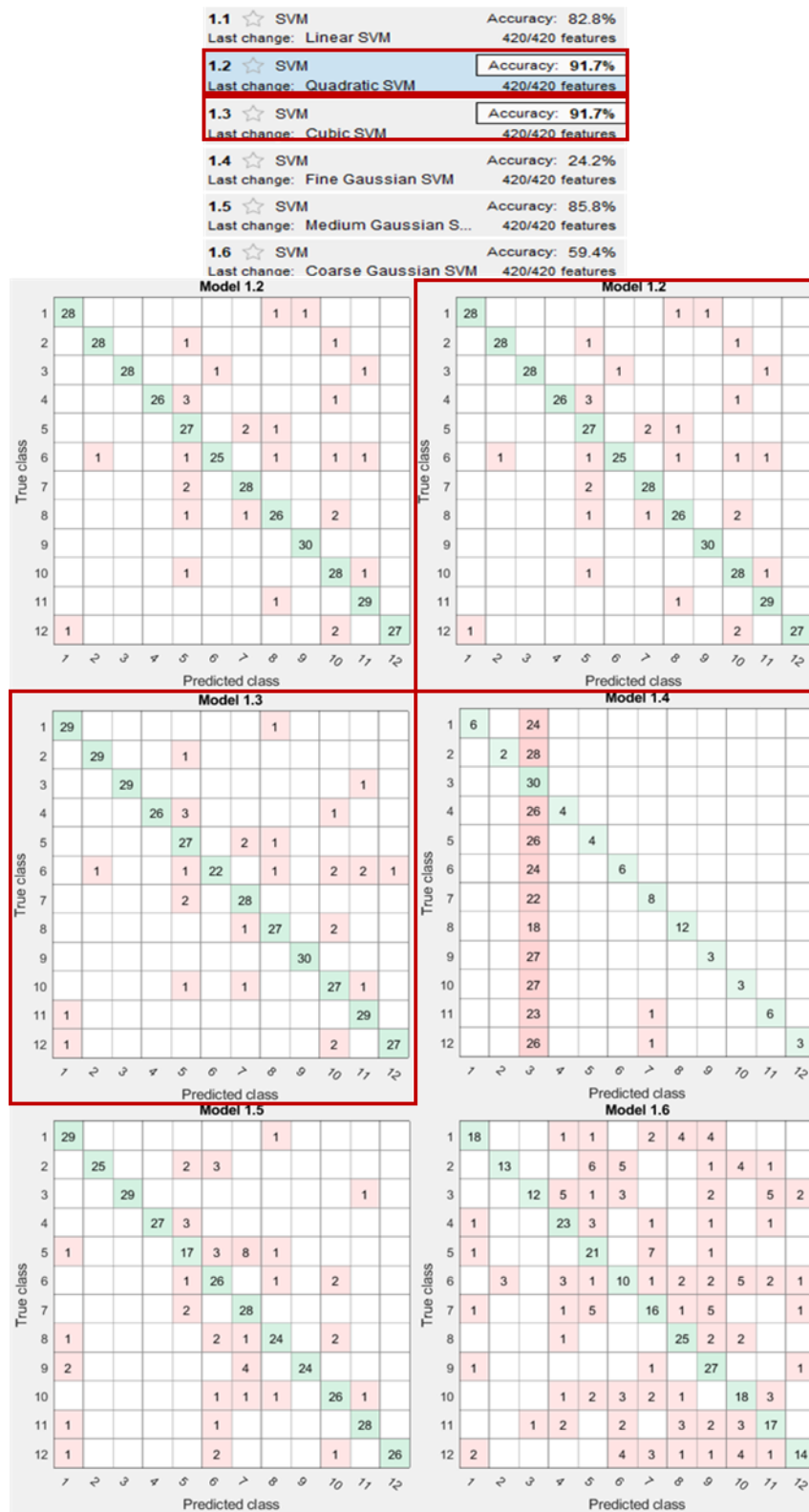


Figura 48. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 11 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2) y el tercero (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

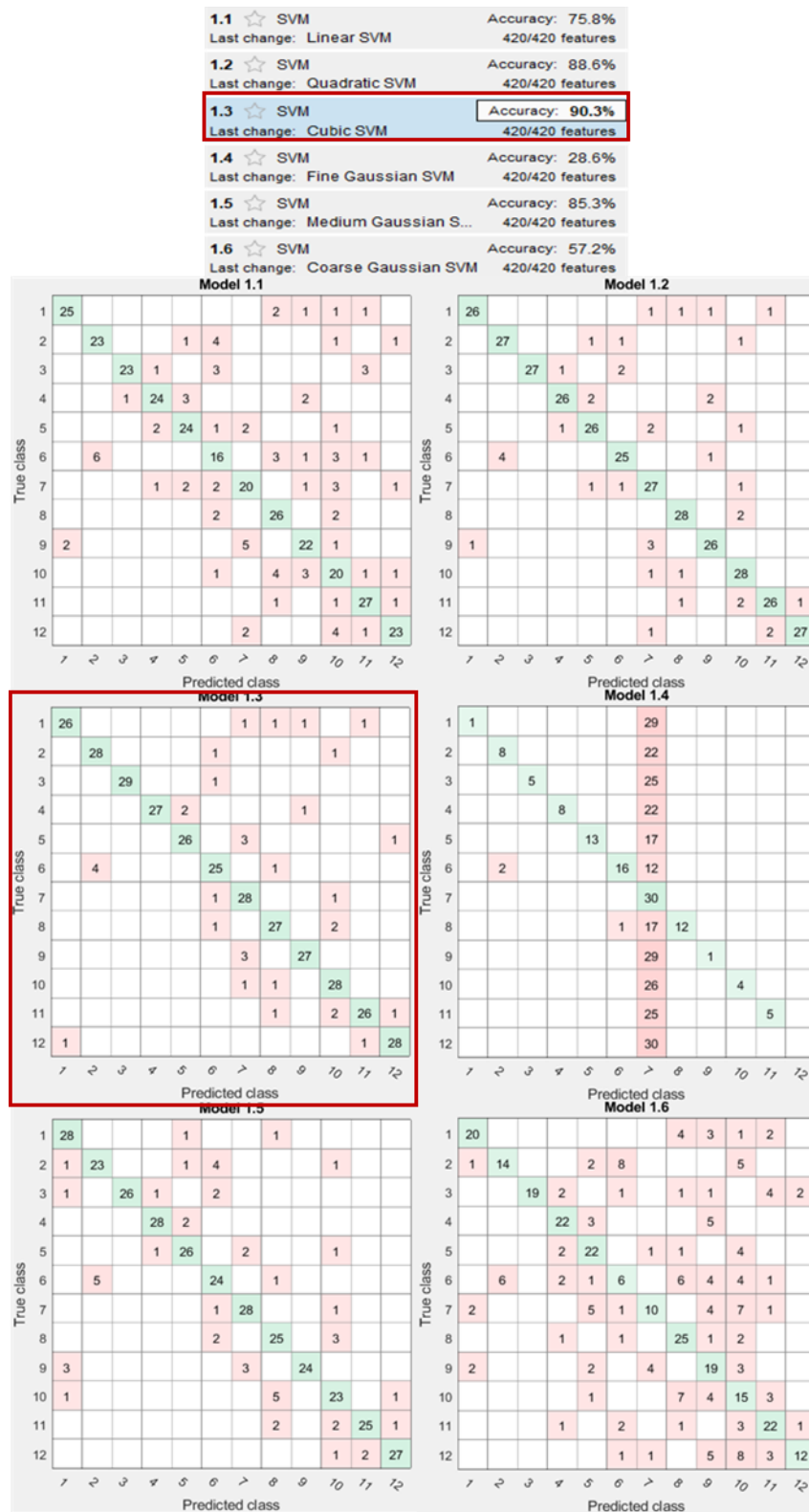


Figura 49. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 12 donde el tercer kernel aplicado (1.3 Cubic SVM, Model 1.3), entrega una mayor numero de aciertos.

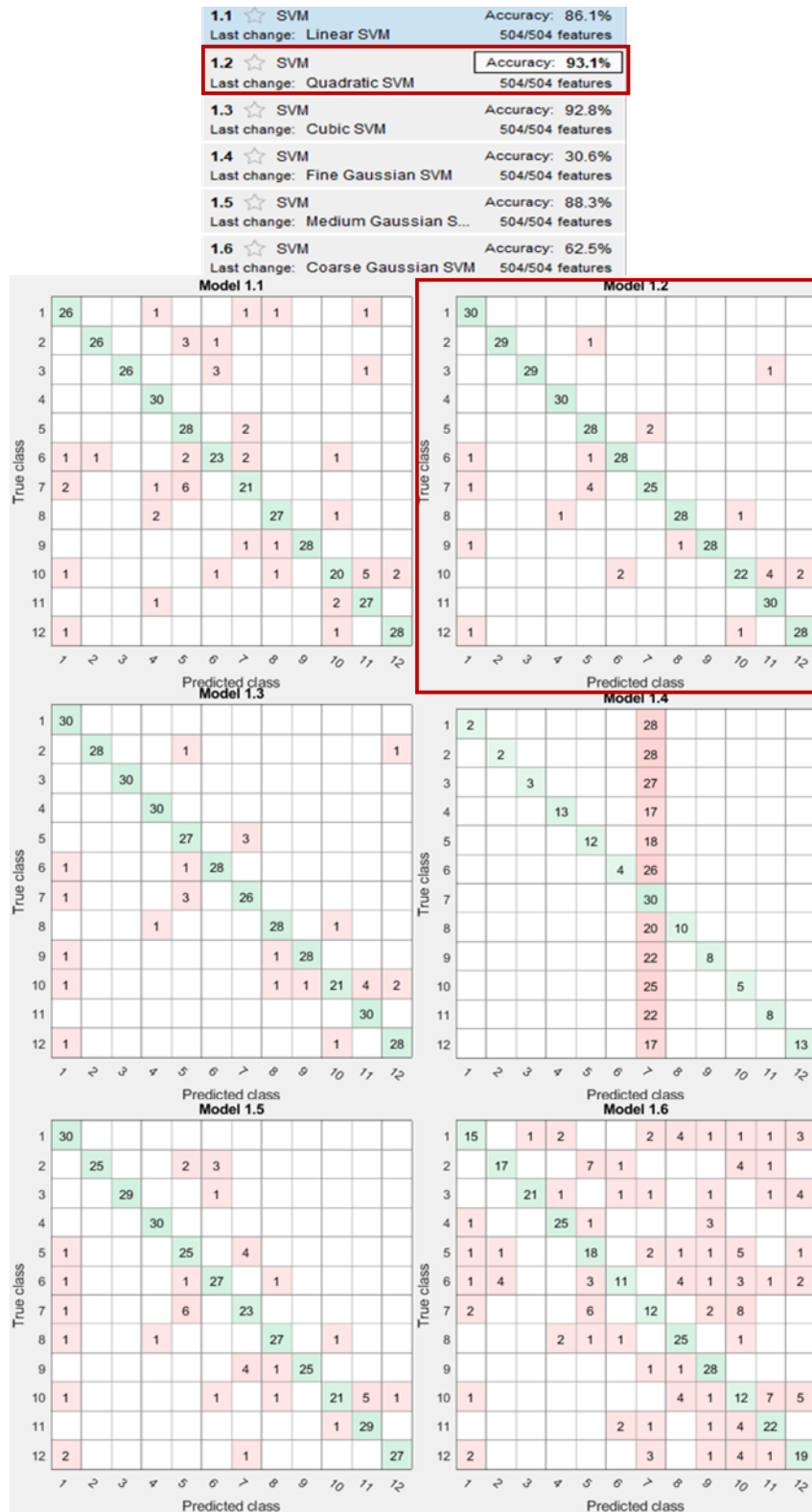


Figura 50. Porcentaje de aciertos y Matrices de confusión obtenidas de la app para la combinación 15 donde el segundo kernel aplicado (1.2 Cuadratic SVM, Model 1.2), entrega una mayor numero de aciertos.

9.2 Resultados obtenidos para Redes Neuronales¹⁸

Confusion Matrix													
Output Class	1	2	3	4	5	6	7	8	9	10	11	12	
	93 7.8%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	2 0.2%	1 0.1%	1 0.1%	1 0.1%	93.0% 7.0%
	0 0.0%	95 7.9%	1 0.1%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	97.9% 2.1%
	1 0.1%	0 0.0%	93 7.8%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	3 0.3%	0 0.0%	1 0.1%	3 0.3%	1 0.1%	90.3% 9.7%
	0 0.0%	0 0.0%	1 0.1%	98 8.2%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	98.0% 2.0%
	0 0.0%	1 0.1%	0 0.0%	1 0.1%	95 7.9%	0 0.0%	4 0.3%	2 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	92.2% 7.8%
	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	91 7.6%	0 0.0%	0 0.0%	0 0.0%	8 0.7%	2 0.2%	0 0.0%	89.2% 10.8%
	1 0.1%	0 0.0%	0 0.0%	1 0.1%	3 0.3%	1 0.1%	87 7.2%	1 0.1%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	91.6% 8.4%
	0 0.0%	2 0.2%	3 0.3%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	91 7.6%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	92.9% 7.1%
	4 0.3%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	6 0.5%	0 0.0%	95 7.9%	0 0.0%	0 0.0%	0 0.0%	89.6% 10.4%
	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.4%	0 0.0%	2 0.2%	0 0.0%	84 7.0%	5 0.4%	1 0.1%	86.6% 13.4%
	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	1 0.1%	0 0.5%	6 7.3%	88 8.0%	0 0.0%	90.7% 9.3%
	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 0.2%	0 0.0%	2 0.2%	0 0.0%	0 0.0%	97 8.1%	95.1% 4.9%
	93.0% 7.0%	95.0% 5.0%	93.0% 7.0%	98.0% 2.0%	95.0% 5.0%	91.0% 9.0%	87.0% 13.0%	91.0% 9.0%	95.0% 5.0%	84.0% 16.0%	88.0% 12.0%	97.0% 3.0%	92.3% 7.8%
Target Class													

Figura 51. Matriz de confusión obtenida de la app para la combinación 15 de la tabla1 usando FFT.

Confusion Matrix													
Output Class	1	2	3	4	5	6	7	8	9	10	11	12	
	97 8.1%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	98.0% 2.0%
	0 0.0%	97 8.1%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	98.0% 2.0%
	1 0.1%	0 0.0%	99 8.3%	0 0.0%	0 0.0%	0 0.0%	4 0.3%	1 0.1%	1 0.1%	1 0.1%	0 0.0%	0 0.0%	92.5% 7.5%
	0 0.0%	0 0.0%	0 0.0%	100 8.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	0 0.0%	0 0.0%	0 0.0%	0 0.0%	95 7.9%	0 0.0%	3 0.3%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	96.0% 4.0%
	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	96 8.0%	0 0.0%	2 0.2%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	96.0% 4.0%
	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 0.3%	0 0.0%	91 7.6%	0 0.0%	1 0.1%	1 0.1%	0 0.0%	0 0.0%	94.8% 5.2%
	0 0.0%	1 0.1%	1 0.1%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	93 7.8%	0 0.0%	0 0.0%	1 0.1%	1 0.1%	94.9% 5.1%
	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	98 8.2%	0 0.0%	0 0.0%	1 0.1%	97.0% 3.0%
	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 0.3%	0 0.0%	3 0.3%	0 0.0%	92 7.7%	0 0.0%	1 0.1%	92.9% 7.1%
	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	5 8.2%	98 8.2%	2 0.2%	92.5% 7.5%
	1 0.1%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	94 7.8%	97.9% 2.1%
	97.0% 3.0%	97.0% 3.0%	99.0% 1.0%	100% 0.0%	95.0% 5.0%	96.0% 4.0%	91.0% 9.0%	93.0% 7.0%	98.0% 2.0%	92.0% 8.0%	98.0% 2.0%	94.0% 6.0%	95.8% 4.2%
Target Class													

Figura 52. Matriz de confusión obtenida de la app para la combinación 15 de la tabla1 usando Wavelet.

¹⁸ Para este caso solo se utiliza la combinación 15 de la tabla 1.

Confusion Matrix													
Output Class	1	2	3	4	5	6	7	8	9	10	11	12	
	94 7.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 0.2%	0 0.0%	2 0.2%	1 0.1%	0 0.0%	1 0.1%	94.0%
	0 0.0%	97 8.1%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	99.0%
	0 0.0%	0 0.0%	96 8.2%	1 0.1%	0 0.0%	2 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	97.0%
	1 0.1%	0 0.0%	0 0.0%	96 8.0%	2 0.2%	0 0.0%	2 0.2%	1 0.1%	0 0.0%	0 0.0%	3 0.3%	0 0.0%	91.4%
	0 0.0%	0 0.0%	0 0.0%	1 0.1%	94 7.8%	0 0.0%	10 0.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	89.5%
	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	90 7.5%	0 0.0%	2 0.2%	0 0.0%	2 0.2%	1 0.1%	0 0.0%	93.8%
	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 0.3%	0 0.0%	86 7.2%	0 0.0%	2 0.2%	0 0.0%	2 0.2%	1 0.1%	91.5%
	2 0.2%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	4 0.3%	0 0.0%	94 7.8%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	92.2%
	2 0.2%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	95 7.9%	0 0.0%	0 0.0%	0 0.0%	96.9%
	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	3 0.3%	0 0.0%	2 0.2%	0 0.0%	91 7.6%	1 0.1%	2 0.2%	91.0%
	0 0.0%	0 0.0%	2 0.2%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	3 0.3%	92 7.7%	1 0.1%	92.0%
	1 0.1%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 0.3%	1 0.1%	95 7.9%	94.1%
Target Class													
													94.0%
													6.0%
													3.0%
													97.0%
													8.6%
													89.5%
													10.5%
													93.8%
													6.3%
													91.5%
													8.5%
													92.2%
													7.8%
													96.9%
													3.1%
													91.0%
													9.0%
													92.0%
													8.0%
													94.1%
													5.9%
													93.5%
													6.5%

Figura 53. Matriz de confusión obtenida de la app para la combinación 15 de la tabla1 usando STFT.

9.3 Resultados obtenidos sobre el Análisis de componentes

9.3.1 Resultados para FFT

- Al aplicar PCA a el arreglo de características que contiene todos los sensores como se observa en la figura 54 (combinación 15, tabla 1), se observa que las características¹⁹ (eje x) que mayor relevancia tiene pueden ser resumidas a 20, pero no indica cuales, por tanto, debe hacerse varias combinaciones para el entrenamiento nuevamente para buscar el mejor resultado.

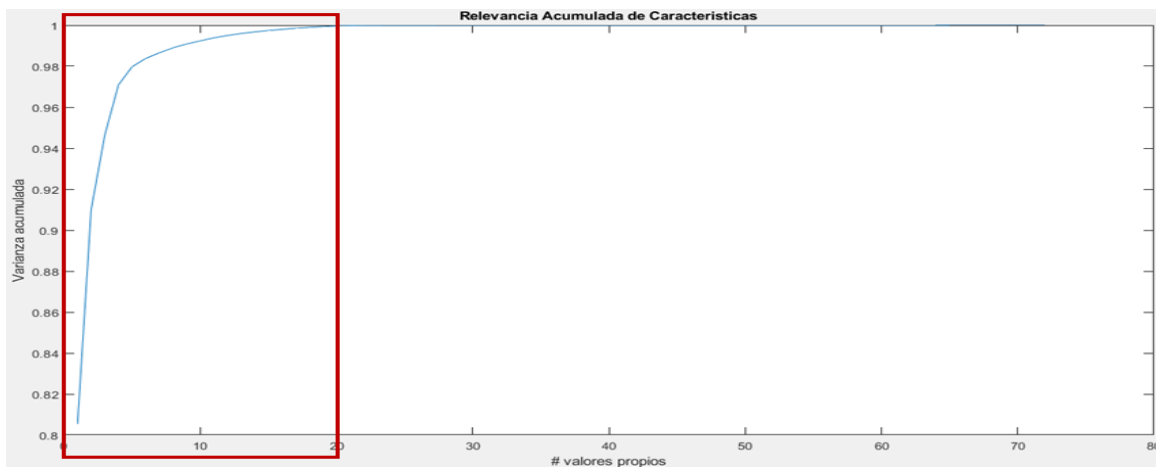


Figura 54. Resultado aplicando PCA a el arreglo de características según la combinación 15 de la tabla 1.

¹⁹Observando la figura 26 si se tienen en cuenta todos los sensores se pueden obtener 72 características, 12 para el acelerómetro, 12 para el giroscopio, 16 para el magnetómetro y 32 para los de electromiografía.

- Aplicando Relief a el arreglo de características que contiene todos los sensores como se observa en la figura 55 (combinación 15, tabla 1), se observa que las características (eje x) tienen asignados algunos pesos que a partir de la muestra 60 son negativos, por tanto, dichas características pueden ser suprimidas para el nuevo entrenamiento.

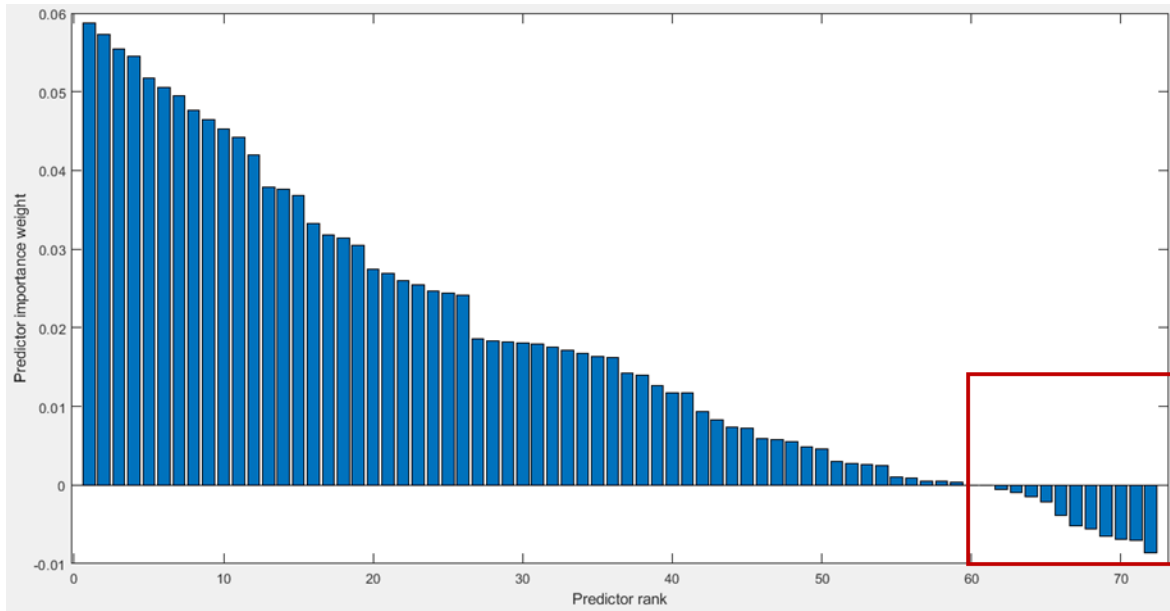


Figura 55. Resultado aplicando Relief a el arreglo de características según la combinación 15 de la tabla 1.

- Aplicando Correlación a el arreglo de características que contiene todos los sensores como se observa en la figura 56 (combinación 15, tabla 1), se presentan franjas donde hay similitudes y tienden a 1 (color amarillo). Lo que indica que muchas de las características en los recuadros pueden ser suprimidas para el entrenamiento, dichas características están contenidas entre 1 a 13, 27 a 40 y 56 a 72.

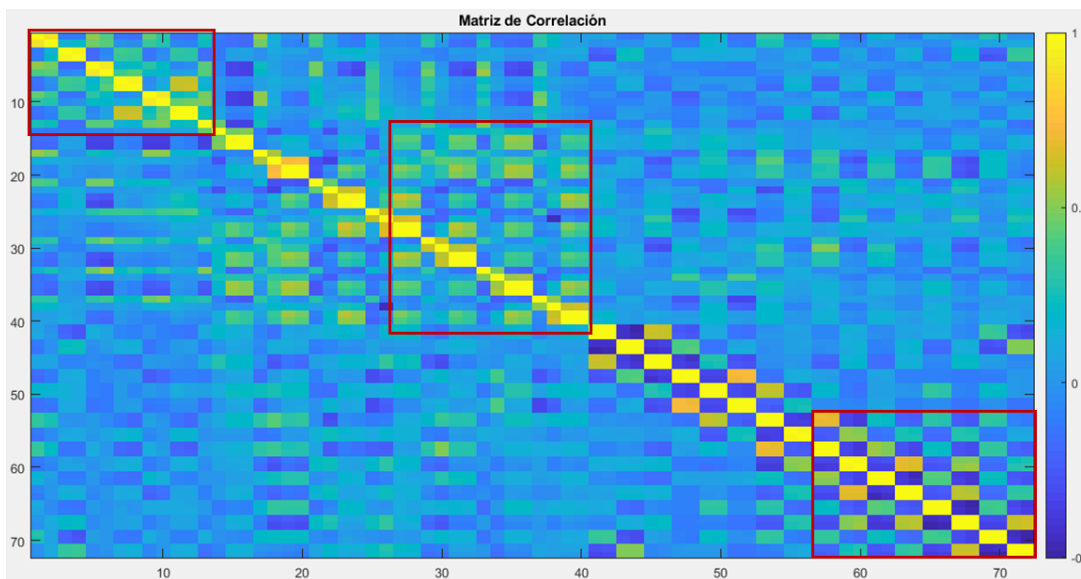


Figura 56. Resultado aplicando Correlación a el arreglo de características según la combinación 15 de la tabla 1.

9.3.2 Resultados para Wavelet

- Al aplicar PCA a el arreglo de características que contiene todos los sensores como se observa en la figura 57 (combinación 15, tabla 1), se observa que las características²⁰ (eje x) que mayor relevancia tiene pueden ser resumidas a 50, pero no indica cuales, por tanto, debe hacerse varias combinaciones para el entrenamiento nuevamente para buscar el mejor resultado.

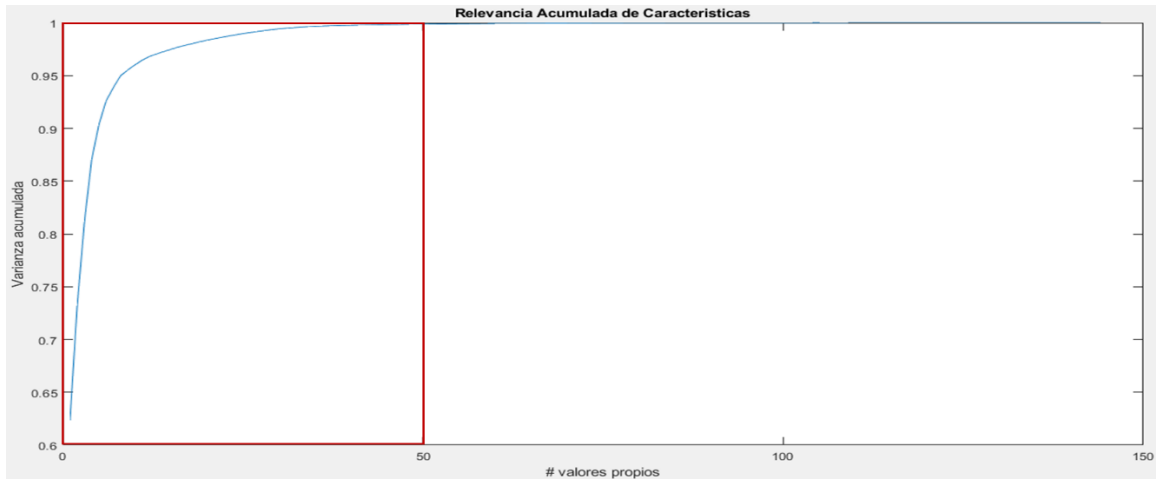


Figura 57. Resultado aplicando PCA a el arreglo de características según la combinación 15 de la tabla 1.

- Aplicando Relieff a el arreglo de características que contiene todos los sensores como se observa en la figura 58 (combinación 15, tabla 1), se observa que las características (eje x) tienen asignados algunos pesos que a partir de la muestra 95 son negativos, por tanto, dichas características pueden ser suprimidas para el nuevo entrenamiento.

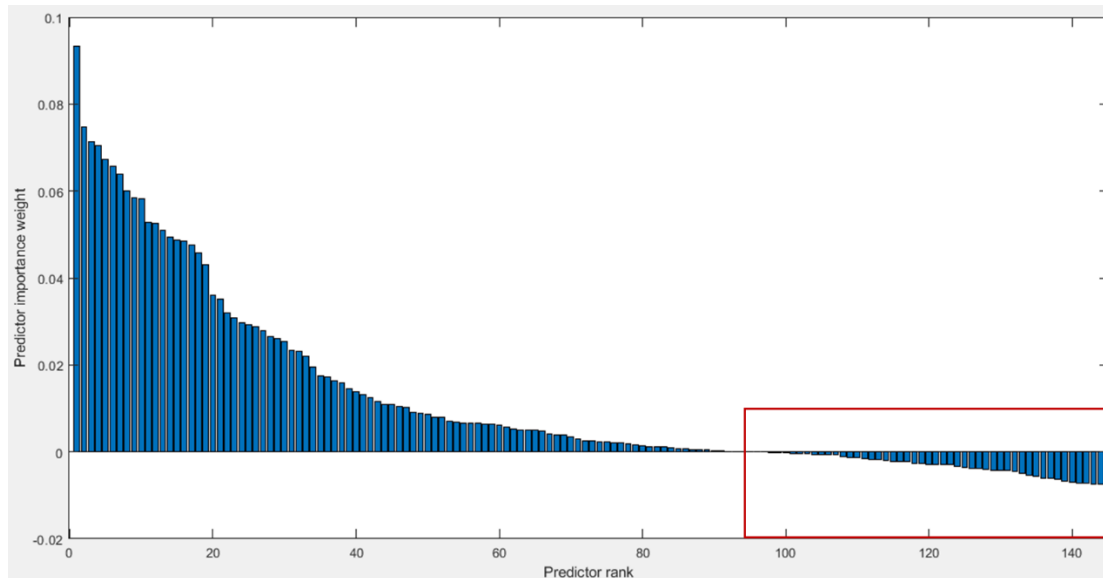


Figura 58. Resultado aplicando Relieff a el arreglo de características según la combinación 15 de la tabla 1.

²⁰Teniendo en cuenta que la Transformada Wavelet entrega una respuesta compuesta por dos arreglos la cantidad de características se duplica por tanto no se tienen en cuenta 72 si no 144.

- Aplicando Correlación a el arreglo de características que contiene todos los sensores como se observa en la figura 59 (combinación 15, tabla 1), se presentan franjas donde hay similitudes y tienden a 1 (color amarillo). Lo que indica que muchas de las características en los recuadros pueden ser suprimidas para el entrenamiento, dichas características están contenidas entre 1 a 23, 82 a 94, 106 a 118 y 133 a 144.

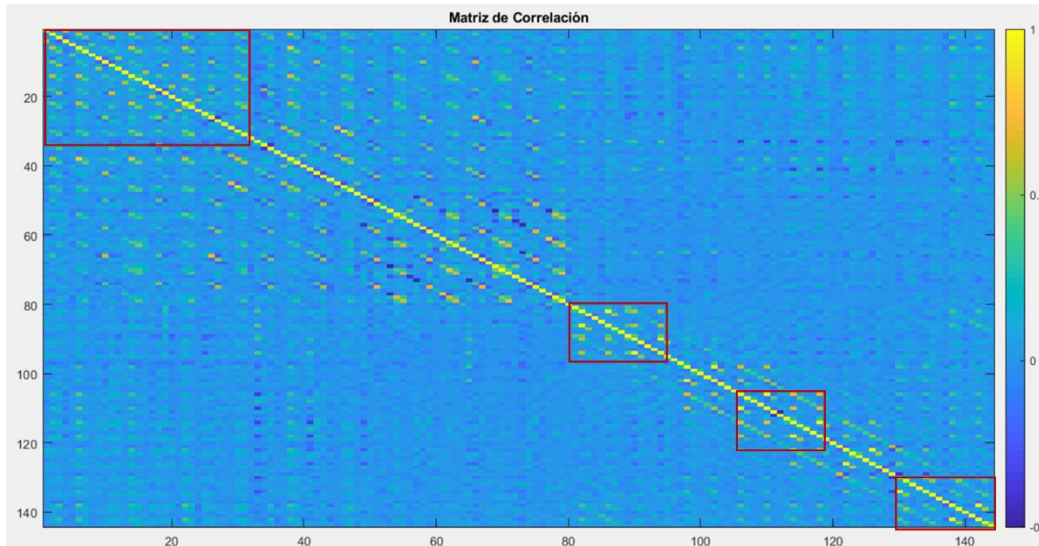


Figura 59. Resultado aplicando Correlación a el arreglo de características según la combinación 15 de la tabla 1.

9.3.3 Resultados para STFT

- Al aplicar PCA a el arreglo de características que contiene todos los sensores como se observa en la figura 60 (combinación 15, tabla 1), se observa que las características²¹ (eje x) que mayor relevancia tiene pueden ser resumidas a 66, pero no indica cuales, por tanto, debe hacerse varias combinaciones para el entrenamiento nuevamente para buscar el mejor resultado.

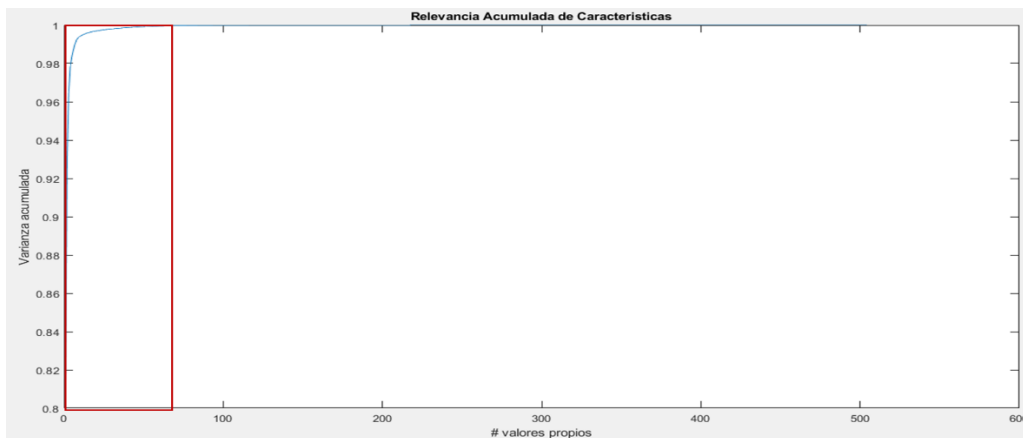


Figura 60. Resultado aplicando PCA a el arreglo de características según la combinación 15 de la tabla 1.

²¹Teniendo en cuenta que la Transformada Corta de Fourier utilizada fue dividida en 7 ventanas la cantidad de características son 504.

- Aplicando Relief a el arreglo de características que contiene todos los sensores como se observa en la figura 61 (combinación 15, tabla 1), se observa que las características (eje x) tienen asignados algunos pesos que a partir de la muestra 381 son negativos, por tanto, dichas características pueden ser suprimidas para el nuevo entrenamiento.

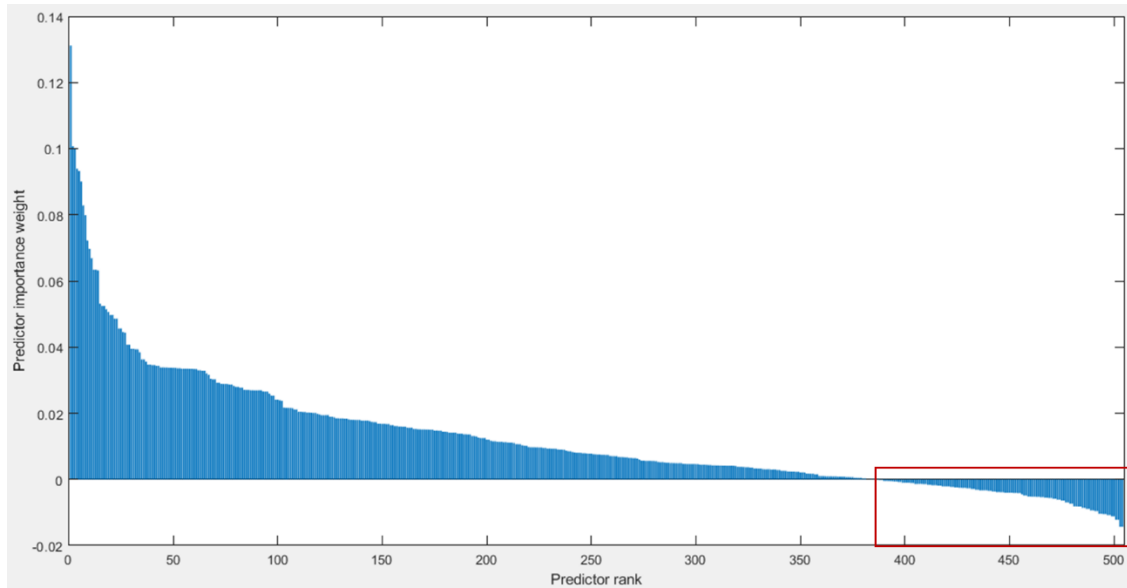


Figura 61. Resultado aplicando Relief a el arreglo de características según la combinación 15 de la tabla 1.

- Aplicando Correlación a el arreglo de características que contiene todos los sensores como se observa en la figura 62 (combinación 15, tabla 1), se presentan franjas donde hay similitudes y tienden a 1 (color amarillo). Lo que indica que muchas de las características en los recuadros pueden ser suprimidas para el entrenamiento, dichas características están contenidas entre 1 a 278, 281 a 321 y 366 a 403.

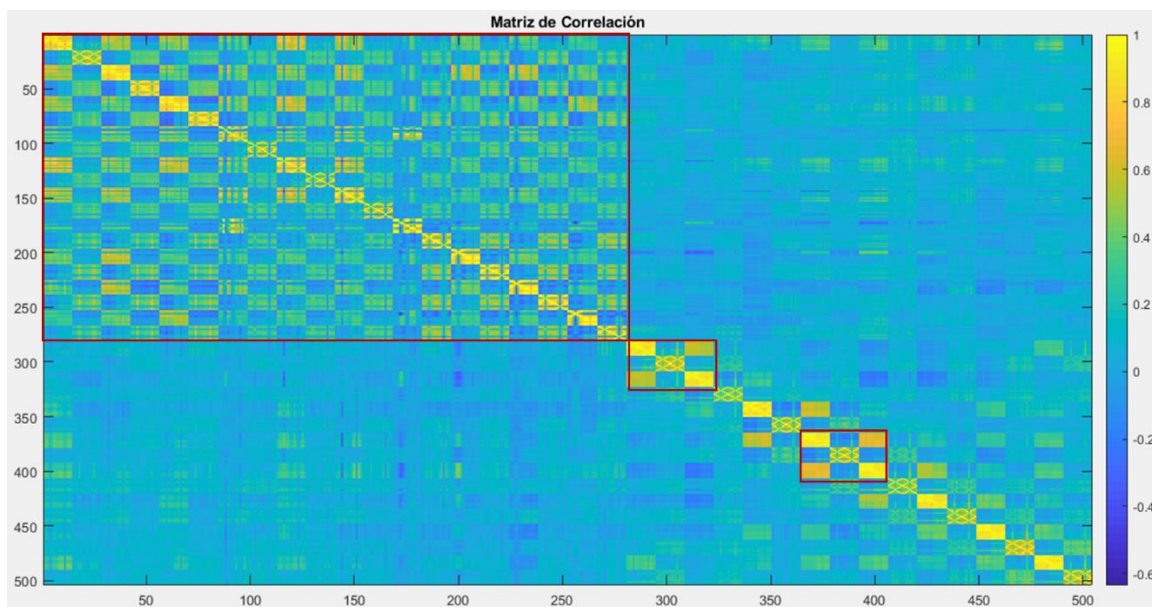


Figura 62. Resultado aplicando Correlación a el arreglo de características según la combinación 15 de la tabla 1.

9.4 Comparación de resultados de Aprendizaje Maquina antes y después del análisis de componentes

Luego del análisis de componentes se procede a quitar características a los diferentes arreglos para el entrenamiento de Aprendizaje Maquina como se muestra a continuación.

9.4.1 Antes y después para FFT aplicando SVM y Redes Neuronales.²²

En la figura 63 se observa como el porcentaje de aciertos para los 6 kernel cambia luego de eliminar las características mayores a la 60 según lo indica los resultados de Relieff, además se muestra las respectivas matrices de confusión para los mejores casos.

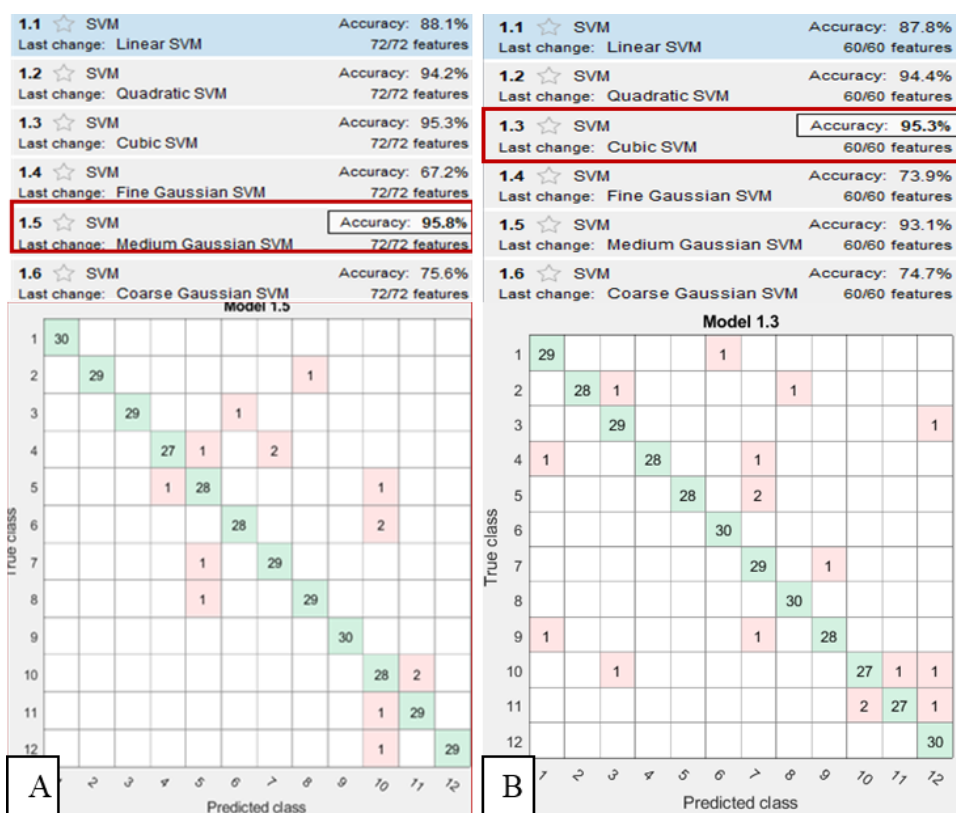


Figura 63. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 40 (95.8%) y B son los nuevos resultados obtenidos (95.3%).

En la figura 64 se observa como el porcentaje de aciertos para los 6 kernel cambia luego de eliminar las características 1 a 13, 27 a 40 y 56 a 72 según lo indica los resultados de la Correlación, además se muestra las respectivas matrices de confusión para los mejores casos.

²² Este análisis solo se hizo para la combinación 15 de la tabla 1.

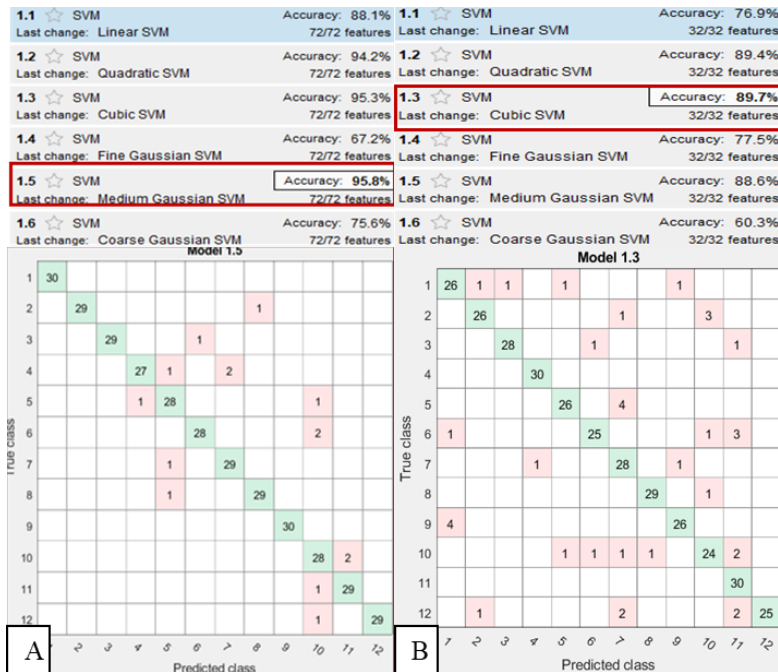


Figura 64. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 40 (95.8%) y B son los nuevos resultados obtenidos (89.7%).

En la figura 65 se observa las matrices de confusión con sus respectivos porcentajes de acierto resultante de aplicar Redes Neuronales las cuales cambian luego de eliminar las características mayores a la 60 según lo indica los resultados de Relieff.

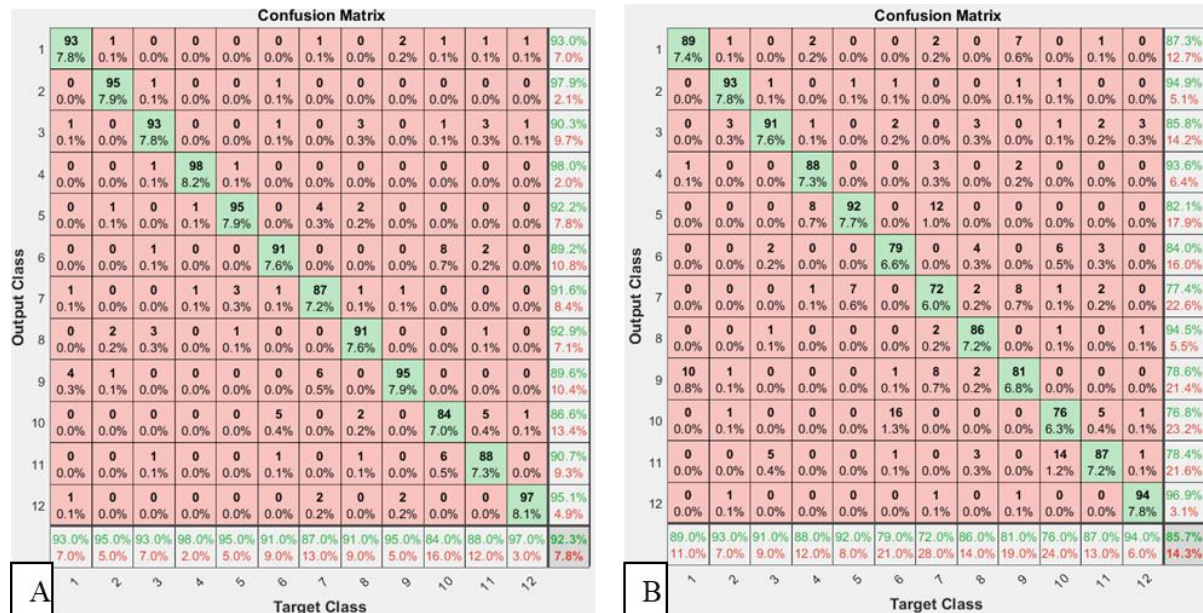


Figura 65. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 51 (92.3%) y B son los nuevos resultados obtenidos (85.7%).

En la figura 66 se observa las matrices de confusión con sus respectivos porcentajes de acierto resultante de aplicar Redes Neuronales las cuales cambian luego de eliminar las características entre 1 a 13, 27 a 40 y 56 a 72 según lo indica los resultados de la Correlación.

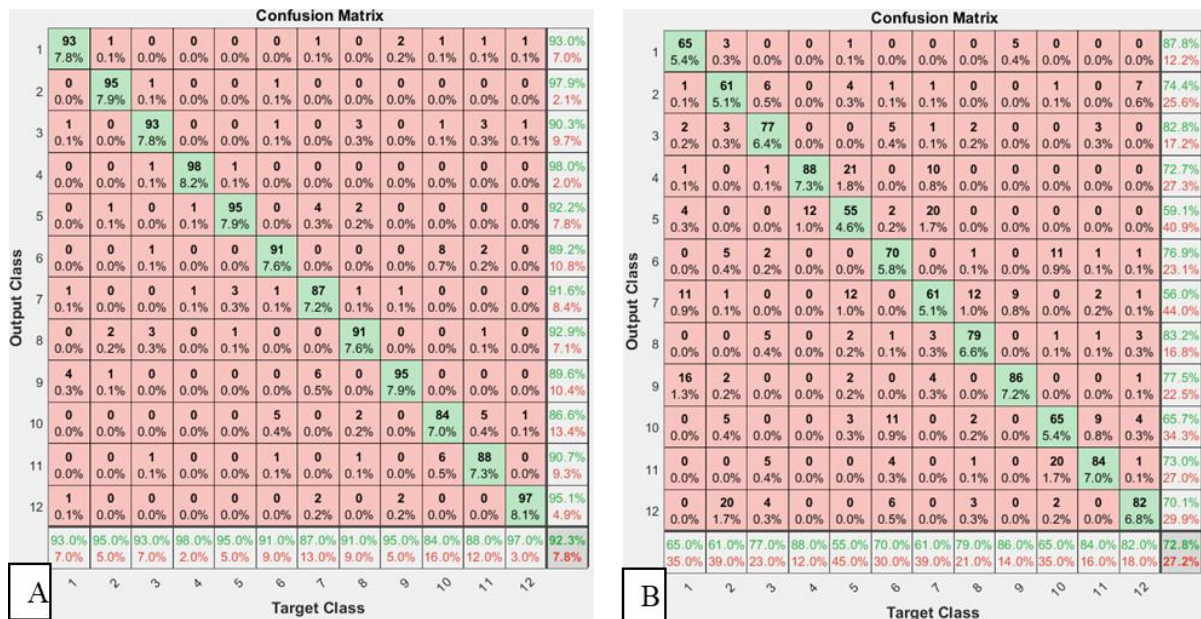


Figura 66. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 51 (92.3%) y B son los nuevos resultados obtenidos (72.8%).

9.4.2 Antes y después para STFT aplicando SVM y Redes neuronales

En la figura 67 se observa como el porcentaje de aciertos para los 6 kernel cambia luego de eliminar las características mayores a la 381 según lo indica los resultados de Relieff, además se muestra las respectivas matrices de confusión para los mejores casos.

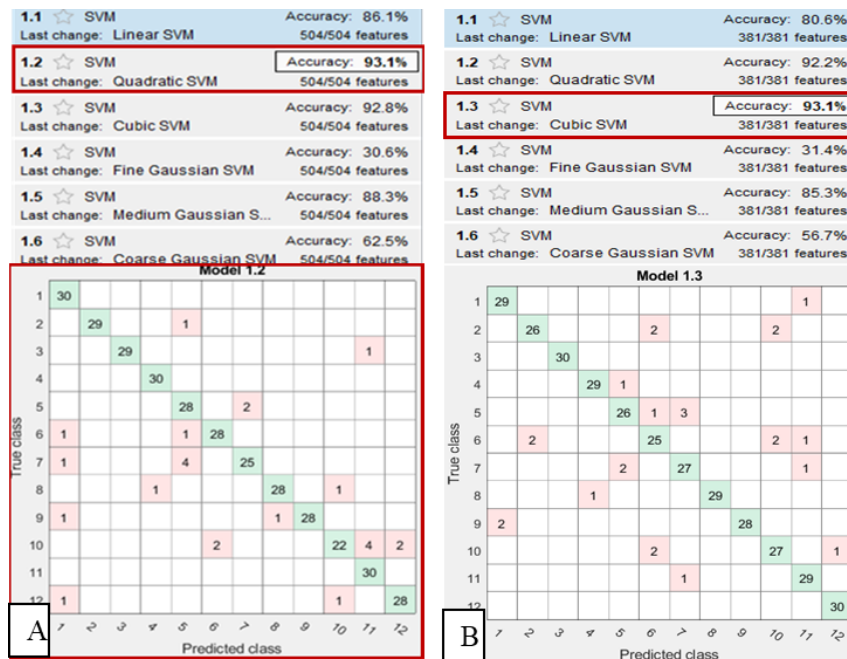


Figura 67. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 50 (93.1%) y B son los nuevos resultados obtenidos (93.1%).

En la figura 68 se observa como el porcentaje de aciertos para los 6 kernel cambia luego de eliminar las características entre 1 a 278, 281 a 321 y 366 a 403 según lo indica los resultados

de la Correlación, además se muestra las respectivas matrices de confusión para los mejores casos.

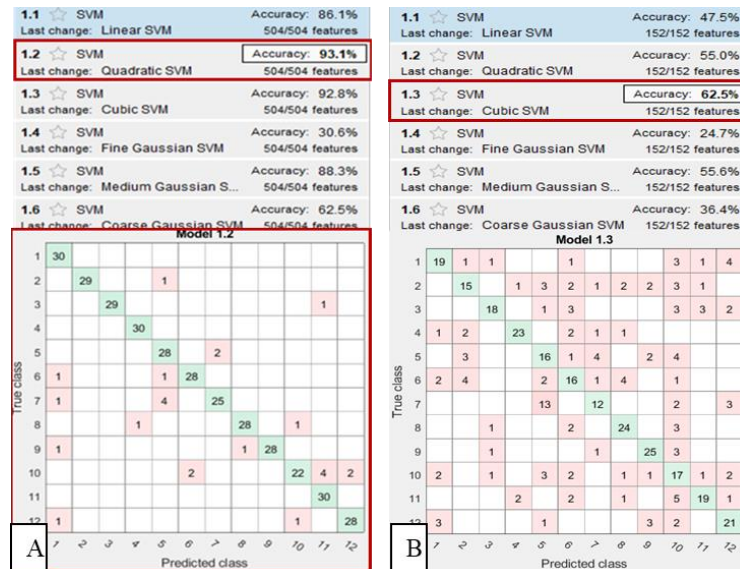


Figura 68. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 50 (93.1%) y B son los nuevos resultados obtenidos (62.5%).

En la figura 69 se observa las matrices de confusión con sus respectivos porcentajes de acierto resultante de aplicar Redes Neuronales las cuales cambian luego de eliminar las características mayores a la 381 según lo indica los resultados de Relief.

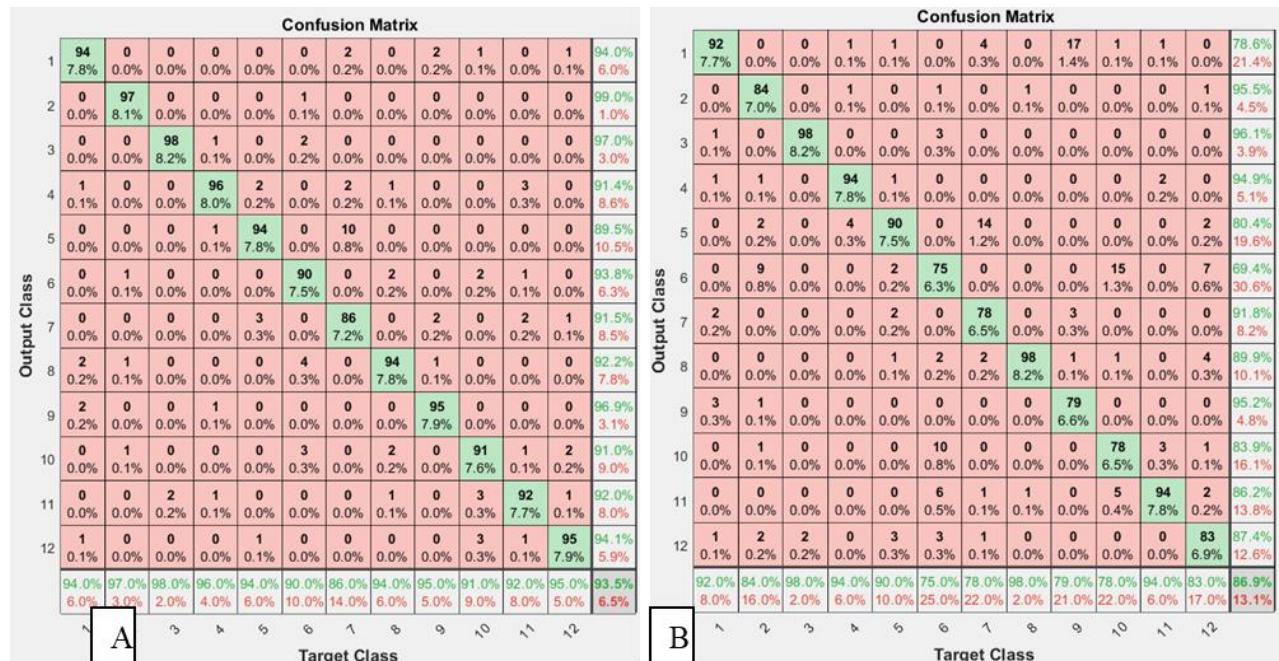


Figura 69. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 53 (93.5%) y B son los nuevos resultados obtenidos (86.9%).

En la figura 66 se observa las matrices de confusión con sus respectivos porcentajes de acierto resultante de aplicar Redes Neuronales las cuales cambian luego de eliminar las características entre 1 a 278, 281 a 321 y 366 a 403 según lo indica los resultados de la Correlación.

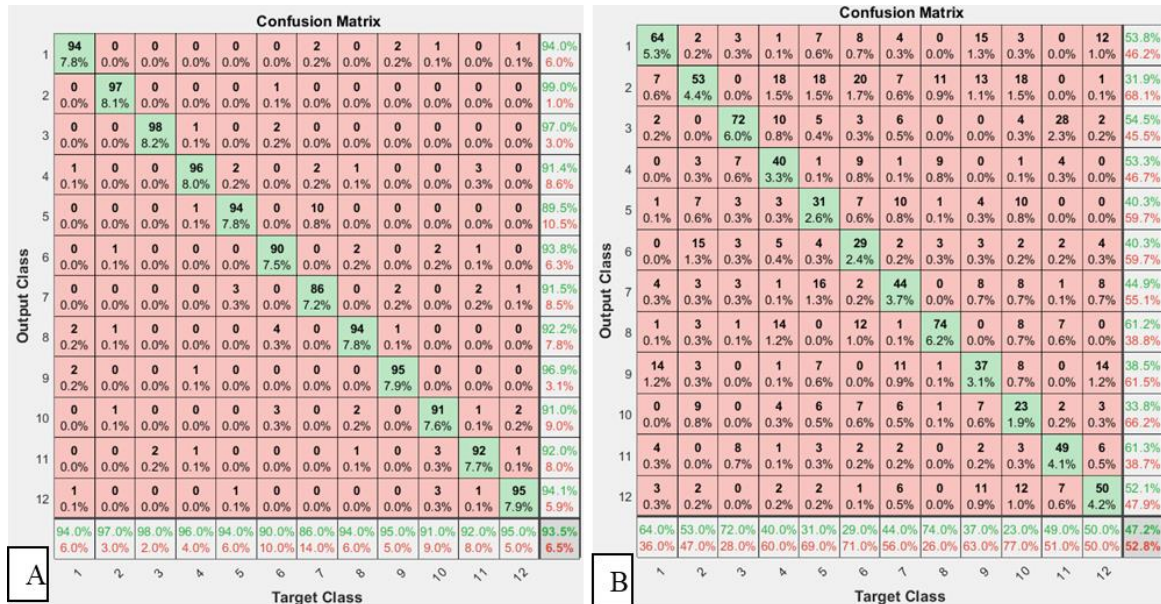


Figura 70. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 51 (93.5%) y B son los nuevos resultados obtenidos (47.2%).

9.4.3 Antes y después para Wavelet aplicando SVM y Redes neuronales

En la figura 71 se observa como el porcentaje de aciertos para los 6 kernel cambia luego de eliminar las características mayores a la 95 según lo indica los resultados de Relieff, además se muestra las respectivas matrices de confusión para los mejores casos.

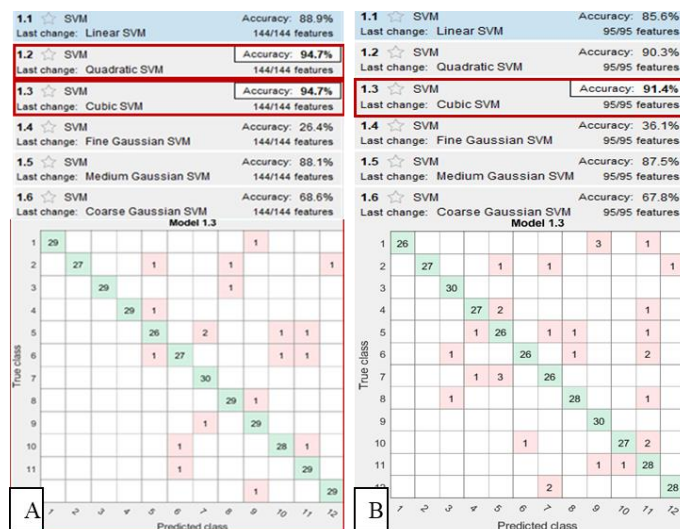


Figura 71. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 45 (94.7%) y B son los nuevos resultados obtenidos (91.4%).

En la figura 72 se observa como el porcentaje de aciertos para los 6 kernel cambia luego de eliminar las características entre 1 a 23, 82 a 94, 106 a 118 y 133 a 144 según lo indica los resultados de la Correlación, además se muestra las respectivas matrices de confusión para los mejores casos.

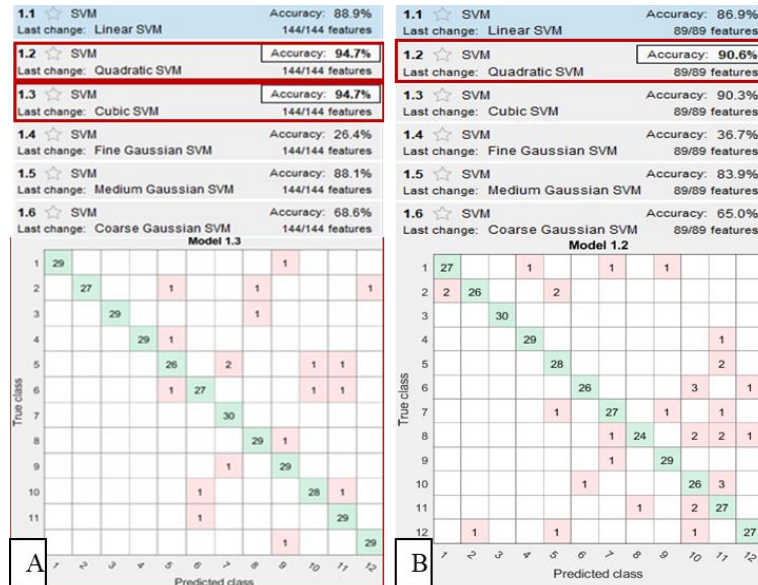


Figura 72. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 45 (94.7%) y B son los nuevos resultados obtenidos (90.6%).

En la figura 73 se observa las matrices de confusión con sus respectivos porcentajes de acierto resultante de aplicar Redes Neuronales las cuales cambian luego de eliminar las características mayores a la 95 según lo indica los resultados de Relieff.

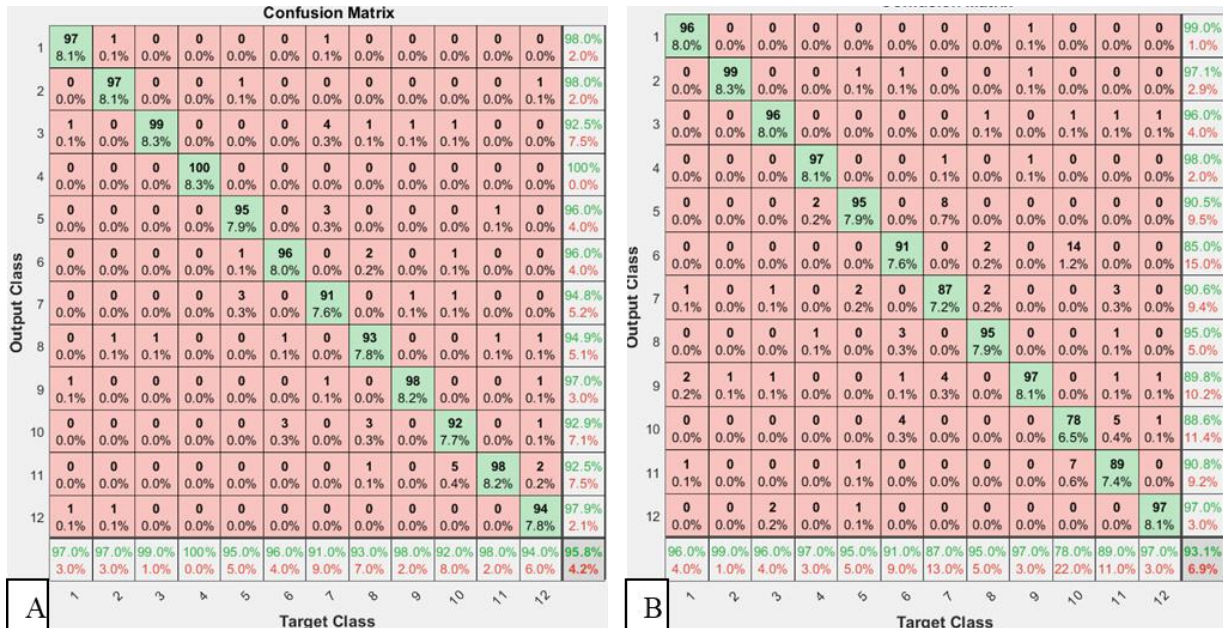


Figura 73. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 52 (95.8%) y B son los nuevos resultados obtenidos (93.1%).

En la figura 74 se observa las matrices de confusión con sus respectivos porcentajes de acierto resultante de aplicar Redes Neuronales las cuales cambian luego de eliminar las características entre 1 a 23, 82 a 94, 106 a 118 y 133 a 144 según lo indica los resultados de la Correlación.

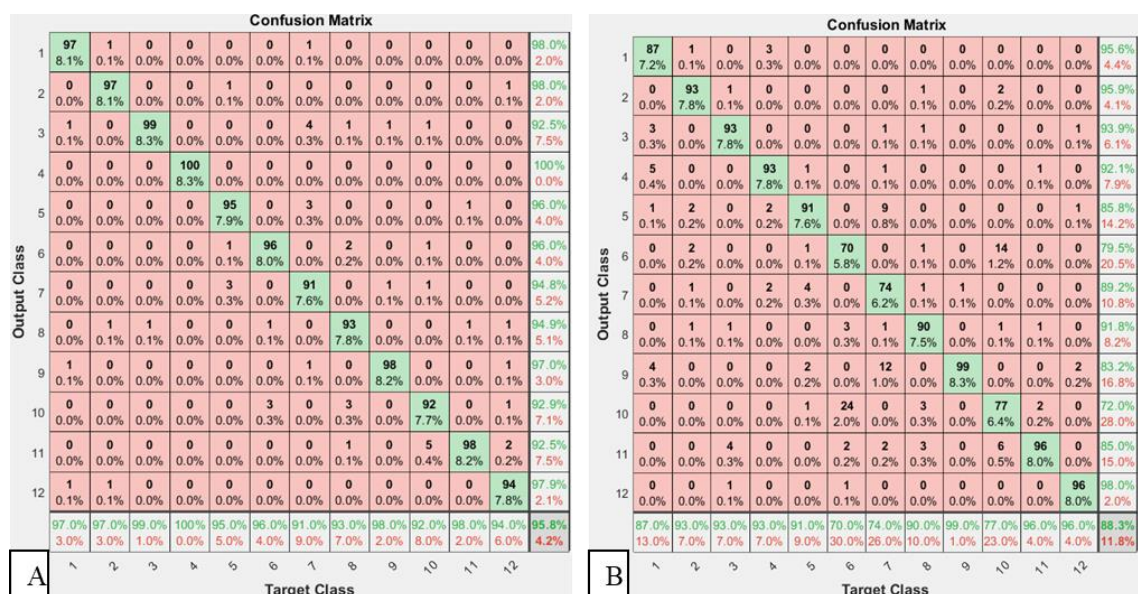


Figura 74. Resultados antes y después de la selección de componentes, donde A son los mejores resultados mostrados en la figura 52 (95.8%) y B son los nuevos resultados obtenidos (88.3%).

10. CONCLUSIONES

- Utilizando los descriptores iniciales con SVM se obtuvieron porcentajes de acierto en rangos entre 78.9% y el 94.7% donde la combinación 3 (Magnetómetro) y 15 (Todos los sensores) son comunes para la FFT, STFT y Wavelet.
- Una vez realizados todos los experimentos se observa que la combinación de todos los sensores asegura un porcentaje de acierto mayor al 90%, en comparación con las posibles combinaciones que se pueden tener de los mismos, cuyos porcentajes de acierto tienden a ser variables e incluso tener un porcentaje menor al 80%
- Al utilizar Redes neuronales siempre se obtuvieron resultados con porcentajes de acierto mayores 92.3% para todos los descriptores, por tanto, marcan una mayor clasificación en comparación con las variaciones de kernel de la SVM.
- Observando los resultados obtenidos a partir del método Relieff y correlación es posible la selección de los datos que no aportan información relevante en el momento de realizar el entrenamiento, por otro lado, PCA indica cuantas características son necesarias para el mismo.
- Teniendo en cuenta los porcentajes de acierto entregados por todos los kernel de SVM, se puede deducir que los momentos estadísticos del sensor Magnetómetro fueron las características más importantes, debido a que entregan mayor información, lo que se traduce en mayor porcentaje de acierto para la clasificación.
- La transformada corta de Fourier y Wavelet para este caso solo presenta una mayor cantidad de datos para el análisis, que no se ve reflejado en el aumento de porcentaje de aciertos en la clasificación, en comparación con la FFT, esto se observa en el hecho de que los porcentajes de acierto de las tres transformadas al aplicar SVM siempre son mayores al 90%.

11. TRABAJOS FUTUROS

- Implementación de un dispositivo que cuente con los mismos sensores vistos, con mano de obra colombiana para bajar costos de producción y permitir la adquisición del mismo por parte de la comunidad sorda.
- Recolectar una base de datos con todas las señas alojadas en el diccionario de señas colombiano para entrenar un algoritmo que permita la traducción en tiempo real de una conversación.
- Llevar el desarrollo a un aplicativo móvil que, por medio de bluetooth permita la clasificación de señas se hecha desde diferentes dispositivos.
- Desarrollar un programa a nivel social que permita la implementación de un clasificador de señas en lugares estratégicos para la inclusión social.

12. BIBLIOGRAFÍA

[1] ELTIEMPO. Estuvo prohibida y cambia por país: curiosidades de la lengua de señas. Publicado el 21 de febrero de 2018 a las 8:34 a.m. [En línea]. <<https://www.eltiempo.com/vida/educacion/curiosidades-de-la-lengua-de-senas-185256>> [Citado el 22 de noviembre de 2018].

[2] INSOR. Diccionario básico. [En línea] <http://www.insor.gov.co/descargar/diccionario_basico_completo.pdf> [Citado el 20 de noviembre de 2018].

[3] Fenascal. INFORME GENERAL DE GESTIÓN 2017. Publicado el 22 de febrero de 2018. [En línea]. <https://fenascal.org.co/images/dian/INFORME_DE_GESTI%C3%93N_2017_-_FINAL-min.pdf> [Citado el 10 de octubre de 2019].

[4] INSOR. Instituto Nacional para Sordos. [En línea]. <<http://www.insor.gov.co/home/>> [Citado el 10 de octubre de 2019]

[5] UTP. Formación en Lenguaje de Señas Colombiana. Publicado el 20 de agosto de 2013. [En línea]. <<https://www.utp.edu.co/vicerrectoria/responsabilidad-social/formacion-en-lengua-de-senas-colombiana.html>> [citado el 11 de octubre de 2019]

[6] RCNRadio. La comunidad sorda supera cada vez más el límite del silencio. Publicado el 16 de julio de 2018. [En línea]. <<https://www.rcnradio.com/recomendado-del-editor/la-comunidad-sorda-supera-cada-vez-mas-el-limite-del-silencio>> [Citado el 4 de abril de 2019].

[7] DANE. Censo Nacional. [En línea]. <<https://www.dane.gov.co/index.php/estadisticas-por-tema/demografia-y-poblacion/censo-nacional-de-poblacion-y-vivenda-2018/cuantos-somos>> [Citado el 3 de abril de 2019].

[8] ANTV. Television con lenguaje para todos. Publicado el 5 de abril de 2016. [En línea]. <<https://antv.gov.co/index.php/prensa/noticias/item/412-television-con-lenguaje-para-todos>> [Citado 11 de noviembre de 2019].

[9] Centro de Relevo. TIC y Discapacidad. [En línea]. <<https://centroderelevo.gov.co/632/w3-channel.html>> [Citado el 8 de noviembre de 2019].

[10] Organización Mundial de la Salud. Sordera y pérdida de la audición. Publicado el 15 de marzo de 2019. [En línea]. <<https://www.who.int/es/news-room/fact-sheets/detail/deafness-and-hearing-loss>> [Citado el 8 de noviembre de 2019].

[11] BBC. Google sign language AI turns hand gestures into speech. Publicado el 20 de agosto de 2019. [En línea]. <<https://www.bbc.com/news/technology-49410945>> [Citado el 12 de noviembre de 2019].

[12] IEE Xplore. Real-Time Malaysian Sign Language Translation using Colour Segmentation and Neural Network. Publicado el 25 de junio de 2007. [En línea]. <<https://ieeexplore.ieee.org/abstract/document/4258110>> [Citado el 12 de noviembre de 2019].

[13] Castilla Acosta Karla Margarita, Betanzos Martínez Omar Eduardo, Jerónimo Bañuelos Julián, Domínguez García Marco. Sistema traductor de lenguaje de señas mexicanas. Publicando el 10 de noviembre de 2017. [En línea]. <[http://www.ecorfan.org/bolivia/researchjournals/Tecnologias de la Informacion/vol4num 13/Revista de Tecnologias de la Informacion V4 N13 7.pdf](http://www.ecorfan.org/bolivia/researchjournals/Tecnologias%20de%20la%20Informacion/vol4num13/Revista%20de%20Tecnologias%20de%20la%20Informacion%20V4%20N13%207.pdf)> [Citado el 12 de noviembre de 2019].

[14] HETAH. [En línea]. <<http://hetah.net/>> [Citado el 10 de abril de 2019]

[15] D. J. Botina-Monsalve, M. A. Domínguez-Vásquez, C. A. Madrigal-González y A. E. Castro-Ospina. Clasificación automática de las vocales en el lenguaje colombiano de

señas. Publicado el 13 de diciembre de 2017. [En línea]. <<https://revistas.itm.edu.co/index.php/tecnologicas/article/view/730/705>> [Citado el 12 de noviembre de 2019].

[16] Rodríguez Rocha Claudia Patricia, Pinares Arias Jhon Alexander, Sánchez Villamil Diego. Prototipo traductor de señales manuales a texto legible, utilizando Kinect. Publicado el 29 de octubre de 2013. [En línea]. <<https://revistas.unilibre.edu.co/index.php/avances/article/view/2749/2169>> [12 de noviembre de 2019].

[17] Sierra Herrera Julián David. Diseño e implementación de un algoritmo para interpretar las vocales del lenguaje de señas colombiano a través de sensores. Publicado en 2017. [En línea]. <<https://repository.usta.edu.co/handle/11634/9181>> [Citado el 12 de noviembre de 2019].

[18] Realpe Fresneda Gustavo Alejandro. Reconocimiento del Lenguaje de Señas Manuales con el Kinect. Publicado en julio 2013. [En línea]. <<https://repositorio.uniandes.edu.co/bitstream/handle/1992/12206/u671097.pdf?sequence=1>> [Citado el 12 de noviembre de 2019].

[19] LSE FÁCIL. Parámetros Formativos del Signo. [En línea]. <<http://lsefacil.es/parametros-formativos-del-signo/>> [Citado el 12 de noviembre de 2019].

[20] SHOWLEAP TECHNOLOGIES. Parámetros formacionales de la LSE: ¿cómo se construyen los signos? Publicado el 1 de diciembre de 2014. [En línea]. <<http://blog.showleap.com/2014/12/parametros-formacionales-lse/>> [Citado el 12 de noviembre de 2019].

[21] NORTH. Getting started with your Myo armband. [En línea]. <<https://support.getmyo.com/hc/en-us/articles/202657596>> [Citado el 25 abril de 2018].

[22] IEEE Xplore. Evaluating Sign Language Recognition Using the Myo Armband. Publicado el 21 de julio 2016. <<https://ieeexplore.ieee.org/document/7517255>> [Citado el 29 de abril de 2019].

[23] IEEE Xplore. Gesture Recognition for Indonesian Sign Language Systems (ISLS) Using Multimodal Sensor Leap Motion and Myo Armband Controllers Based-on Naïve Bayes Classifier. Publicado el 18 de enero de 2018. [En línea]. <<https://ieeexplore.ieee.org/document/8262533>> [Citado el 29 de abril de 2019].

[24] IEEE Xplore. Deaf talk using 3D animated sign language: A sign language interpreter using Microsoft's kinect v2. Publicado el 1 de septiembre de 2016. [En línea]. <<https://ieeexplore.ieee.org/document/7556002>> [Citado el 15 de mayo de 2019].

[25] IEEE Xplore. American Sign Language Recognition using Deep Learning and Computer Vision. Publicado el 24 de enero de 2019. [En línea]. <<https://ieeexplore.ieee.org/document/8622141>> [Citado el 18 de mayo de 2019].

[26] IEEE Xplore. Hand Gesture Recognition for Deaf People Interfacing. Publicado el 18 de septiembre de 2006. [En línea]. <<https://ieeexplore.ieee.org/document/1699157>> [Citado el 24 de mayo de 2019].

[27] IEEE Xplore. Mexican sign language recognition using movement sensor. Publicado el 1 de octubre de 2015. [En línea]. <<https://ieeexplore.ieee.org/document/7281531>> [Citado el 29 de mayo de 2019].

[28] IEEE Xplore. Evaluating a Speech Communication System for Deaf People. Publicado el 22 de agosto de 2011. [En línea]. <<https://ieeexplore.ieee.org/document/5993744/>> [Citado el 1 de junio de 2019].

[29] IEEE Xplore. Application design sign language Colombian for mobile devices VLSCApp (Voice Colombian sign language app) 1.0. Publicado el 4 de agosto de 2016. [En línea]. <https://ieeexplore.ieee.org/document/7528378> [Citado el 5 de junio de 2019].

[30] IEEE Xplore. Application to support the process of training sign language through Microsoft Kinect®. Publicado el 23 de noviembre de 2015. [En línea]. <<https://ieeexplore.ieee.org/document/7333462>> [Citado el 15 de junio de 2019].

[31] Méndez Romero Alexander. Ramírez Torres Jorge Edison. DESARROLLO DE UN PROTOTIPO DE SOFTWARE COMO HERRAMIENTA PARA LA ENSEÑANZA DEL LENGUAJE DE SEÑAS COLOMBIANAS. Publicado el 20 de enero 2012. [En línea]. <<http://biblioteca.usbbog.edu.co:8080/Biblioteca/BDigital/66037.pdf>> [Citado el 16 de junio de 2019].

[32] Huircan Juan Ignacio. Conversores Análogo-Digital y Digital-Análogo: Conceptos Básicos. [En línea]. <<https://core.ac.uk/download/pdf/47281302.pdf>>. [Citado el 18 de junio de 2019].

[33] Jesús de la Fuente María. Sensores. [En línea]. <<http://www.isa.cie.uva.es/~maria/sensores.pdf>> [Citado el 20 de junio de 2019]

[34] ResearchGate. Stepping-stones to Transhumanism: An EMG-controlled Low-cost Prosthetic Hand for Academia. Publicado el 10 de septiembre de 2018. [En línea]. <[_____](#)> [Citado el 20 de junio de 2019].

[35] Technical Features and Functionalities of Myo Armband: An Overview on Related Literature and Advanced Applications of Myoelectric Armbands Mainly Focused on Arm Prostheses. Publicado el 11 de junio de 2018 [En línea]. <https://www.researchgate.net/publication/324889539_Technical_Features_and_Functionalities_of_Myo_Armband_An_Overview_on_Related_Literature_and_Advanced_Applications_of_Myoelectric_Armbands_Mainly_Focused_on_Arm_Prostheses> [Citado el 20 de junio de 2019].

[36] EcuRed. Señal eléctrica. [En línea]. <https://www.ecured.cu/Se%C3%B1al_el%C3%A9ctrica> [Citado el 4 de julio de 2019].

[37] Open Course Ware de la Universitat de València. Introducción al Procesado Digital de Señales. Publicado el 2009. [En línea]. <<http://ocw.uv.es/ingenieria-y-arquitectura/1-1/tema1.pdf>> [Citado el 3 de agosto de 2019].

[38] PANAMAHITEK. Señales continuas, analógicas, discretas y digitales. Publicado el 22 de enero de 2015. [En línea]. <<http://panamahitek.com/senales-continuas-analogicas-discretas-y-digitales/>> [Citado el 10 de agosto de 2019].

[39] Alva Coras Carlos Arturo. PROCESAMIENTO DE SEÑALES DE ELECTROMIOGRAFÍA SUPERFICIAL PARA LA DETECCIÓN DE MOVIMIENTO DE DOS DEDOS DE LA MANO. Publicado en 2012. [En línea]. <http://repositorio.urp.edu.pe/bitstream/handle/urp/416/Alva_ca.pdf?sequence=1&isAllowed=y> [Citado el 25 de agosto de 2019].

[40] Ingeniería Mecafenix. Sensor inercial o Sensor IMU. Publicado el 23 de julio de 2018. [En línea]. <<https://www.ingmecafenix.com/automatizacion/sensores/sensor-inercial/>> [Citado el 28 de agosto de 2019].

[41] Vistronica. Imu. [En línea]. <<https://www.vistronica.com/imu/>> [Citado el 28 agosto de 2019].

[42] Cáceres Juan Pablo. Análisis Espectral 1: Transformada Corta de Fourier y Ventanas. Publicado en agosto de 2017. [En línea]. <https://ccrma.stanford.edu/workshops/cm2007/topics/clases/PDFs/07espectral1_handout.pdf> [Citado el 15 de septiembre de 2019].

[43] R. Castro Liliana, M. Castro Silvana. Wavelets y sus aplicaciones. Publicado en 1995. [En línea]. <http://sedici.unlp.edu.ar/bitstream/handle/10915/24289/Documento_completo.pdf?sequence=1> [Citado el 20 de septiembre de 2019].

[44] MathWorks. Fft. Introducido en R2006. [En línea]. <<https://es.mathworks.com/help/matlab/ref/fft.html>> [Citado el 25 de septiembre de 2019].

[45] Gonzales Gonzales Rafael Alberto. Algoritmo basado en Wavelets aplicado a la detección de incendios forestales. Publicado el 14 de mayo de 2010. [En línea].

<http://catarina.udlap.mx/u_dl_a/tales/documentos/mel/gonzalez_g_ra/capitulo3.pdf>

[Citado el 25 de septiembre de 2019].

[46] YouTube. Simon Xu. Easy Introduction to Wavelets. Publicado el 10 de noviembre de 2015 [En línea]. <<https://www.youtube.com/watch?v=ZnmvUCtUAEE>> [Citado el 25 de septiembre de 2019].

[47] Azor Montoya Jesús Rubén. LA TRANSFORMADA WAVELET. Publicado en 2006. [En línea]. <<http://www.um.edu.ar/ojs2019/index.php/RUM/article/view/22/24>> [Citado el 25 de septiembre de 2019].

[48] Cortes O. Jimmy Alexander, Medina A. Francisco Alejandro, Chaves O. José Andrés. DEL ANÁLISIS DE FOURIER A LAS WAVELETS
ANÁLISIS DE FOURIER. Publicado en mayo de 2007. [En línea]. <<https://dialnet.unirioja.es/descarga/articulo/4807129.pdf>> [Citado el 25 de septiembre de 2019].

[49] MathWorks. Stft. Incluido para R2019. [En línea]. <<https://www.mathworks.com/help/signal/ref/stft.html>> [Citado 28 de septiembre de 2019].

[50] Gonzales Kevin. Medidas de tendencia central. Publicado el 5 de abril de 2017. [En línea]. <http://www.kevingonzalez.blogspot.com/2017/04/medidas-de-tendencia-central.html>> [Citado el 28 de septiembre de 2019].

[51] MathWorks. Mean. Incluido en R2006. [En línea]. <<https://la.mathworks.com/help/matlab/ref/mean.html>> [Citado el 29 de septiembre de 2019].

[52] Soporte de Minitab. ¿Qué es la desviación estándar? [En línea]. <<https://support.minitab.com/es-mx/minitab/18/help-and-how-to/statistics/basic-statistics/supporting-topics/data-concepts/what-is-the-standard-deviation/>> [Citado el 30 de septiembre de 2019]

[53] MathWorks. Std. incluido en R2006. [En línea].
<<https://la.mathworks.com/help/matlab/ref/std.html>> [Citado el 30 de septiembre de 2019].

[54] Ecomipedia. Curtosis. [En línea].
<<https://economipedia.com/definiciones/curtosis.html>> [Citado el 30 de septiembre de 2019].

[55] Universo Formulas. Curtosis. Publicado el 25 de noviembre de 2014. [En línea].
<<https://www.universoformulas.com/estadistica/descriptiva/curtosis/>> [Citado el 30 de septiembre de 2019].

[56] MathWorks. Kurtosis. Incluido en R2006. [En línea].
<<https://www.mathworks.com/help/stats/kurtosis.html>> [Citado el 30 de septiembre de 2019].

[57] Álvarez Sebastián. Prezi. EL SESGO ESTADISTICO. [En línea]. Publicado el 8 de abril de 2016. <<https://prezi.com/19wh3bpu0vqy/el-sesgo-estadistico/>> [Citado el 30 de septiembre de 2019].

[58] Ávila María. Sesgo y Curtosis. Publicado el 22 de mayo de 2009. [En línea].
<<http://mariaavila-maria.blogspot.com/2009/05/sesgo-y-curtosis.html>> [Citado el 30 de septiembre de 2019].

[59] MathWorks. Skewness. Incluido en R2006. [En línea].
<<https://www.mathworks.com/help/stats/skewness.html>> [Citado el 30 de septiembre de 2019].

[60] Slide Share. Estadística. MiguelLav. Publicado el 24 de agosto de 2012. [En línea].
<<https://es.slideshare.net/MiguelLav/probabilidad-y-estadistica-teoria>> [Citado el 30 de septiembre de 2019].

[61] Developers. McCrea Nick. Introducción a la Teoría de Aprendizaje de Máquina y sus Aplicaciones: Un Tutorial Visual con Ejemplos. [En línea]. <<https://www.toptal.com/machine-learning/introduccion-a-la-teoria-de-aprendizaje-de-maquina-y-sus-aplicaciones-un-tutorial-visual-con-ejemplos>> [Citado el 30 de septiembre de 2019].

[62] Oracle. ¿Qué es el aprendizaje automático? [En línea]. <<https://www.oracle.com/co/artificial-intelligence/what-is-machine-learning.html>> [Citado el 30 de septiembre].

[63] Zambrano Juan. ¿Aprendizaje supervisado o no supervisado? Conoce sus diferencias dentro del machine learning y la automatización inteligente. Publicado el 30 de marzo de 2018. [En línea]. <<https://medium.com/@juanzambrano/aprendizaje-supervisado-o-no-supervisado-39ccf1fd6e7b>> [Citado el 30 de septiembre].

[64] EcuRed. Funcion Kernel. [En línea]. <https://www.ecured.cu/Funci%C3%B3n_Kernel> [Citado el 30 de septiembre].

[65] Martínez Carlos. WordPress. El Hiperplano en el espacio E4D. Publicado el 25 de agosto de 2016. [En línea]. <<https://gratis840.wordpress.com/2016/08/25/el-hiperplano-en-el-espacio-e4d/>> [Citado el 30 de septiembre de 2019].

[66] Carmona Suarez Enrique J. Tutorial sobre Máquinas de Vectores Soporte (SVM). Publicado el 11 de julio de 2014. [En línea]. <[http://www.ia.uned.es/~ejcarmona/publicaciones/\[2013-Carmona\]%20SVM.pdf](http://www.ia.uned.es/~ejcarmona/publicaciones/[2013-Carmona]%20SVM.pdf)> [Citado el 30 de septiembre de 2019].

[67] Larrañaga Pedro, Inza Iñaki, MoujahidRedes Abdelmalik. Redes Neuronales. [En línea] <<http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t8neuronales.pdf>> [Citado el 30 de septiembre de 2019].

[68] Aler Mur Ricardo. SELECCIÓN Y GENERACIÓN DE ATRIBUTOS-I. [En línea]. <<http://ocw.uc3m.es/ingenieria-informatica/analisis-de-datos/transparencias/SELECCION%20DE%20ATRIBUTOS.pdf>> [Citado el 15 de octubre de 2019].

[69] MathWorks. Relieff. Incluido en R2010. [En línea]. <<https://www.mathworks.com/help/stats/relieff.html>> [Citado el 17 de octubre de 2019].

[70] López José Francisco. Ecomipedia. Varianza. [En línea]. <<https://economipedia.com/definiciones/varianza.html>> [Citado el 18 de octubre de 2019].

[71] Ramon S. Gustavo. Correlación entre variables. [En línea]. <http://viref.udea.edu.co/contenido/menu_alterno/apuntes/ac36-correlacion-variables.pdf> [Citado el 18 de octubre de 2019].

[72] BBC. Las increíbles extremidades del ser humano: manos y pies. Publicado el 2 de marzo de 2014. [En línea]. <https://www.bbc.com/mundo/noticias/2014/03/140227_ciencia_manos_y_pies_jgc_finde> [Citado el 21 de octubre de 2019].

[73] Tchvibes. Thalmic Labs Expands Retail Presence in Canada, Shows Off Myo Armband Uses. Publicado el 19 de agosto de 2015. [En línea]. <<https://techvibes.com/2015/08/19/thalmic-labs-expands-retail-presence-in-canada-shows-off-myo-armband-uses-2015-08-19>> [Citado el 21 de octubre de 2019].

[74] MathWorks. Myo SDK MATLAB MEX Wrapper. Incluido R2018. [En línea]. <<https://www.mathworks.com/matlabcentral/fileexchange/55817-myo-sdk-matlab-mex-wrapper>> [Citado el 22 de octubre de 2019].

[75] MathWorks. Kaiser. Incluido en R2006. [En línea].
<<https://www.mathworks.com/help/signal/ref/kaiser.html>> [Citado el 22 de octubre de 2019].